user has input correct type and/or value of information. If there is some input error, error messages can be prompted to tell the user to input the correct type and/or value. In addition, by using WMLScript, the WML browser can gather the results of a series user interaction and then send the results to the server in a batch.

Device Access: Through WMLScript, the user can access the device's hardware facilities and peripherals, as well as the software functions.

Device Software Extension: Since WMLScript can utilize a device hardware and software facilities, it allows the extension of existing software functionalities by downloading new software components from service providers or device manufacturers. This way, new software functionalities can be added without making changes to the firmware of the device.

15.3 WAP SYSTEM ARCHITECTURE

The WAP system architecture is based on the WWW server/clicent architecture of the Internet. Generally, the architecture is composed of three elements (see Figure 15.2):

Client: It is is the WML browser in a wireless device. It issues WAP requests to a server.

Server: It is the entity which provides services and where resources are located. This can be an ordinary Internet-based server or a WAP-capable server.

Gateway: It provides protocol conversion between the WWW protocol stack and the WAP protocol stack, by using content encoders and decoders. Thus, effectively, a gateway acts as a proxy server. When protocol conversion is performed at the gateway, it can minimize wireless communication overhead at the client side. On the other hand, the gateway can also cache frequently-requested contents so as to reduce the request/response time.

Figure 15.2 shows a detailed view of the WAP infrastructure. A mobile user can access to Internet services provided by an ordinary server or a WAP-capable server. If the server is WAP-capable, then it can directly generate WML content. The WML content will then be sent to the WAP gateway using HTTP. After that, the WAP gateway will convert the WML content into a binary form and then transmit to the mobile device using WAP protocols.

On the contrary, when the mobile user needs to access to an ordinary server which generates only HTML pages, the generated HTML pages will go through a HTML filter first. The function of the HTML filter is to strip away contents that not suitable for presenting in a mobile device, and then convert HTML codes into WML codes. The HTML filter can be collocated with the WAP gateway, or located in another

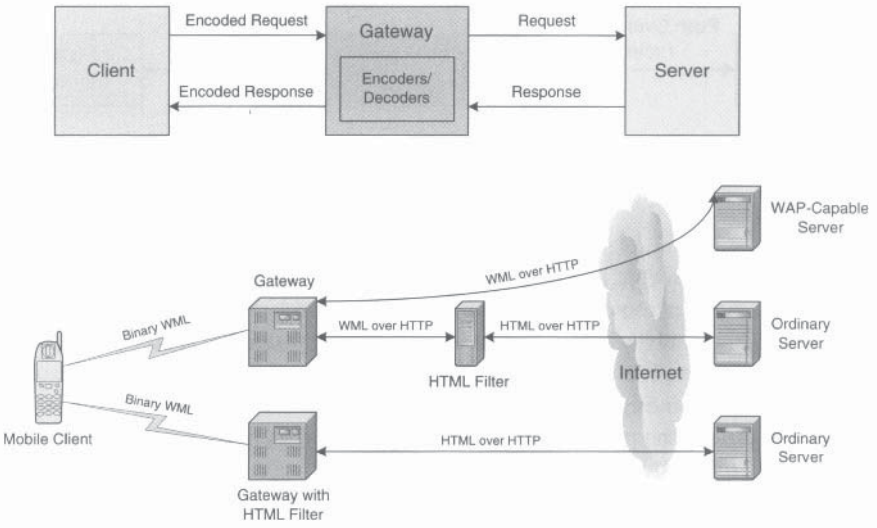


Figure 15.2 WAP infrastructure.

location. If it is located in another location, it will send the filtered WML content to the WAP gateway using HTTP.

Push Architecture

The architecture discussed above is the common pull architecture based on the client/server paradigm, i.e., the client explicitly requests content from the server. The WAP system architecture also specifies a push architecture to enhance the WAP services, in which the server sends messages to the client without explicit request from the client. This push architecture is very useful in delivering messages like instant news, email indication, advertising, etc.

In the push architecture, the server and the gateway are called the push initiator (PI) and the push proxy gateway (PPG), respectively. Figure 15.3 shows the interaction of PI, PPG, and the client. Specifically, the PI sends a push message to the PPG using the push access protocol. The PPG then extracts the client address from the received push message, and checks if the message can be forwarded to the client by finding a mapping of the extracted client address to a format valid in the client's wireless network. If the push message can be forwarded, the PPG then transforms the message to a binary format for efficient transmission to the client using the push over the air (OTA) protocol.

Apart from processing push messages from the PI (i.e., forwarding, cancellation, and replacement of push messages), the PPG also offers other supplementary functionalities to enhance the sending of push messages. These include notifying the PI about the result and status of processing push messages, and telling the PI about the client's capabilities and preferences on content transformation. When a mobile device



Figure 15.3 WAP push architecture.

is not available to accept a push message (e.g., when the device is busy with other tasks or it is at a location of having poor network connection), then the PI can send a short message, namely a service indicator, to the client instead of sending a relatively large push message. The content of the service indicator contains a uniform resource identifier (URI) pointing to a service to be consulted. Consequently, the user can go back to retrieve the push message later by initiating a client request to the server indicating by the URI.

WTA Architecture

To support traditional telephony functionalities in a mobile device, WAP specifies the wireless telephony application (WTA) framework. As illustrated in Figure 15.4, the core of WTA is the WTA user agent. The WTA agent can access to telephony services through a WTA interface (WTAI), providing the user with the ability to make/receive phone calls, access to the phone book, etc. The WTA server provides WTA-related services to the client through the WAP gateway. For example, when there is an incoming voice message, the voice server of the mobile network service provider will indicate the WTA server of the voice message. Then the WTA server may send a push message to the mobile device to notify the user of the new voice message. In the push message, it includes a URL pointing to the server to which the user should contact in order to retrieve the voice message.

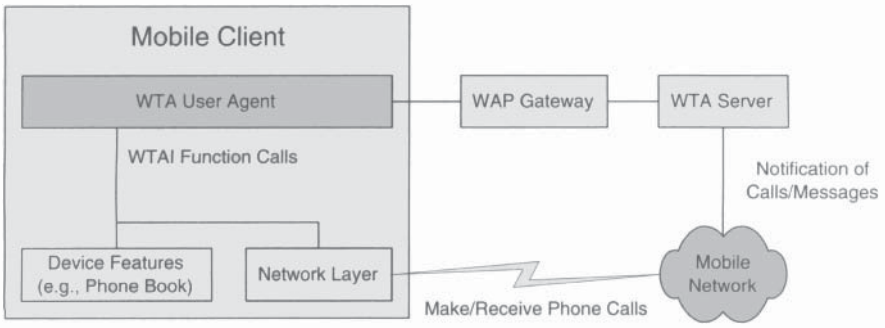


Figure 15.4 WTA architecture.

15.4 WAP PROTOCOL STACK

Overview

Figure 15.5 shows the WAP protocol stack. The lightweight protocol stack is designed with two main goals—to minimize bandwidth requirement and to maximize the number of supported network types. Thus, WAP is designed to be bearer-independent, i.e., it does specify particular bearer services and it tends to use existing data services such as GSM and GPRS. On the other hand, the layered protocol stack is designed to be scalable. Each layer provides a well-defined interface to its upper layer such that each layer can be developed independent of each other. This way, it allows minimum changes to the protocol stack when new bearers or transport protocol stacks are added. It also allows quick addition of new applications and services that are not specified in the WAP specification.

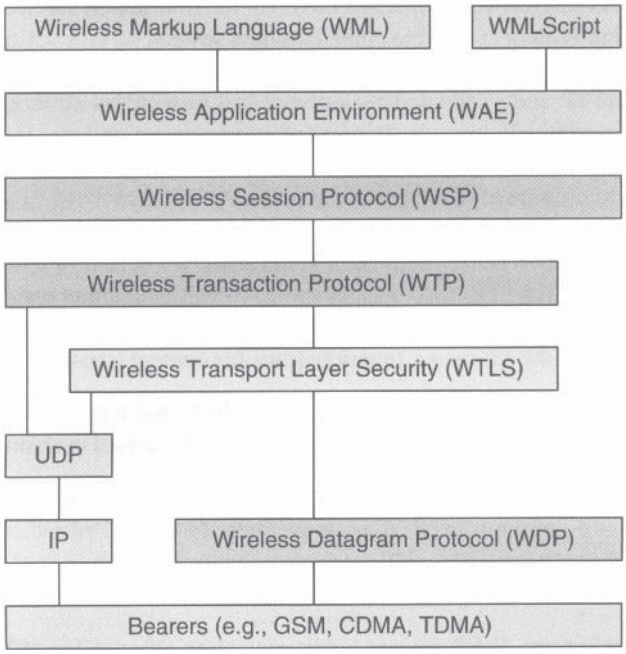


Figure 15.5 The WAP protocol stack.

Wireless Application Environment (WAE)

WAE provides an application framework for mobile devices. It specifies tools and formats for service providers, software developers and hardware manufacturers to develop WAP applications and services. The application environment specified by WAE is based on existing technologies in the WWW paradigm, such as HTML, XML, and JavaScript.

The main elements in WAE are as follows:

WML User Agent: WAE user agents are software executing in mobile devices to support specific functionalities. The WML user agent is a particular user agent that is defined to support WML and WMLScript for content browsing.

WTA User Agent: The WTA user agent is another user agent defined to support telephony services.

Content Generator: It is an application or service residing on ordinary servers. It serves requests from WAP clients and produces standard formats in the response content.

Wireless Session Protocol (WSP)

WSP provides two kinds of session services to its upper layer, namely the connection-oriented session service which operates on top of the transaction service WTP (Wireless Transaction Protocol), and the connectionless session service which operates on top of the datagram service WDP (Wireless Datagram Protocol). Specifically, WSP is based on HTTP and is adapted for optimal use in wireless environments where the data rate is relatively low and there is frequent connection loss. For example, in contrast to HTTP, which is stateless, WSP allows a shared state between a server and a client so as to provide efficient content transfer.

The operation of WSP is based on a request/response paradigm. In addition to this, a server push operation is also defined in WSP to allow a server to send unrequested content to a client. The PDU (protocol data unit) conveyed in either operation consists of a header and a body. In the body, it might contain WML, WMLScript, image data, etc. WSP provides the following functionalities for content transfer:

Session Management: Sessions can be established from a client to a server. Sessions can be long lived, suspended, resumed, and released in an orderly manner.

Capability Negotiation: During session establishment, the client and server can agree on a common level of protocol functionality. Parameters can be negotiated are server/client SDU size, maximum outstanding requests, protocol options, etc.

Content Exchange: WSP defines binary encoding scheme for efficient content exchange between the client and the server in wireless environments.

In WSP, a collection of service primitives are defined to form the interface between WSP and the users of its upper layers in WAE. These primitives have various parameters associated with them, and they form the basis for session management and content exchange. Figure 15.6 shows how session establishment is achieved between the client and the server by using the **S-Connect** primitives. Specifically, the client's WSP user issues an **S-Connect.req** to WSP to request for establishing a session with the WSP user on the server. On issuing the **S-Connect.req**, the parameters passed are

server address (SA), client address (CA), client headers (CH), and requested capabilities (RQ). Then at the server side, **S-Connect.ind** is issued to transfer the parameters from the client to the WSP user. If the WSP user at the server side accepts the request for establishing a session, it responds by issuing **S-Connect.res**, with server headers (SH) and negotiated capabilities (NC) being the parameters. Finally, at the client side, **S-Connect.cnf** is issued to the WSP user to inform it about the negotiated capabilities.

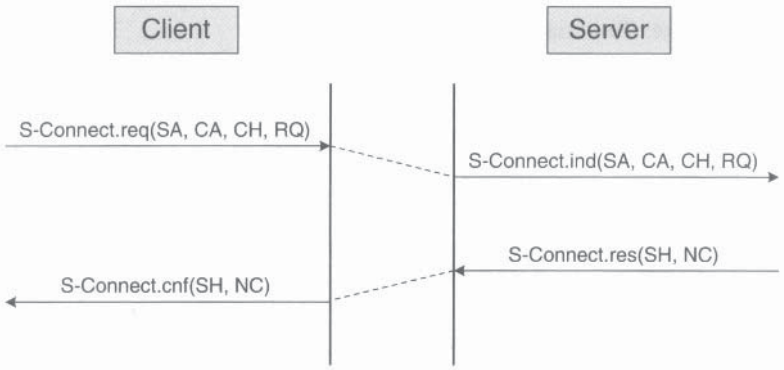


Figure 15.6 WSP session establishment.

Wireless Transaction Protocol (WTP)

WTP is a transaction-oriented protocol, where a transaction is defined to be a request-response pair. Using WTP, there is no explicit connection setup or teardown, which reduces much amount of overhead when compared to TCP. Specifically, WTP provides a reliable connection-less service. Other features of WTP are as follows:

- It achieves reliability by using acknowledgement, retransmission, duplicate removal, and transaction identifier.
- It supports asynchronous transaction, abort of transaction, and report of status of transaction.
- It supports concatenation of messages and delayed acknowledgement so as to reduce the number of messages to be sent out.
- To increase reliability, a WTP user can be requested to confirm the receive of a message.
- There are three classes of transaction services provided, namely Class 0, Class 1, and Class 2. Class 0 provides an unreliable transaction service without the need for a result message, meaning that the transaction is stateless and therefore cannot be aborted. This class of service can be used for providing

unreliable push operations. Class 1 provides a reliable transaction service, also without a result message. It can be used for reliable push operations. The responder of this class of service maintains state information for some time after an ACK has been sent out to handle cases where retransmission of the ACK is needed. Finally, Class 2 provides a reliable transaction service of the classic request/response paradigm.

WTP provides the following three service primitives to its upper layer:

- **TR-Invoke**: To initiate a transaction.
- **TR-Result**: To send back the result of a previously initiated transaction.
- **TR-Abort**: To abort an existing transaction.

Wireless Transport Layer Security (WTLS)

WTLS is on top of WDP to provide, when requested by applications, security services between the mobile device and the WAP gateway or WAP server. WTLS is based on the TLS (Transport Layer Security) protocol, and is optimized for use in low-bandwidth and high-latency wireless networks. It also takes into consideration the relatively low processing power and low memory limitations of mobile devices when choosing cryptographic algorithms. WTLS can provide different levels of security measures to mobile devices: privacy, data integrity, authentication, and denial-of-service protection.

Within WTLS, there are four protocols defined:

Handshake Protocol: It is a four-phase protocol to allow the client and the server to authenticate each other and to negotiate the compression and cryptographic algorithms, and other related security settings used to protect data in WTLS records. A four-phase handshaking is performed between the client and the server before any data can be transmitted.

Change Cipher Spec Protocol: The client/server uses this protocol to notify the other of the change of the cryptographic settings.

Alert Protocol: This protocol is used to notify the other peer entity that an error has occurred, by sending an alert message. An alert message can be of the type warning, critical, or fatal. If it is a fatal message, the current connection will be terminated immediately.

Record Protocol: User data from higher layer protocol (e.g., WTP) is encapsulated in a number of steps using the record protocol.

The details of the steps performed in the record protocol are further elaborated in Figure 15.7 First of all, the user data is compressed using a compression algorithm. Then HMAC is used to calculate a message authentication code (MAC) of the compressed data. The MAC is then appended to the compressed data. After that, the

concatenated data is encrypted using a symmetric encryption algorithm, and finally a record protocol header is prepended to the encrypted data.

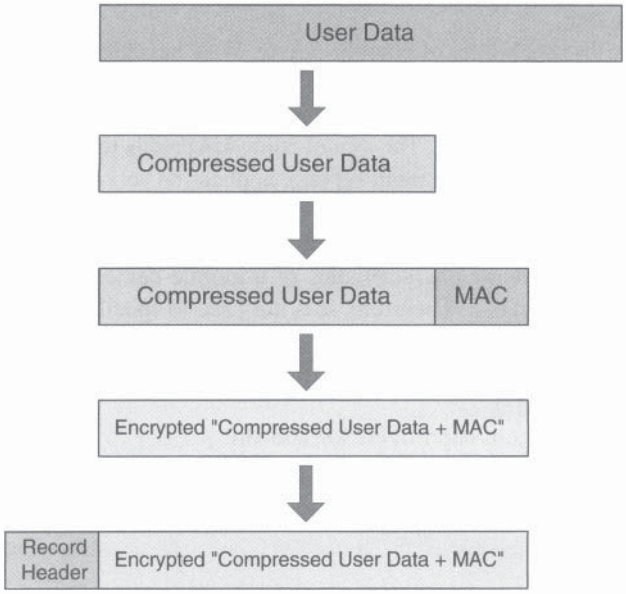


Figure 15.7 Steps performed in the WTLS record protocol.

Wireless Datagram Protocol (WDP)

WDP provides an adaptation on top of many different bearer services to support datagram transport service to higher layers in the WAP protocol stack. WDP is designed to be bearer-independent and thus the adaptation needed depends on the services provided by the target bearer. If the bearer supports IP services, then UDP can be used as WDP, and adaptation is minimal. Otherwise, WDP may need to support functions like data segmentation and reassembly.

During WDP datagram transmission, if errors occur (e.g., destination unreachable), WAP specifies the use of wireless control message protocol (WCMP) for error handling. WCMP is much like the ICMP in IP for diagnostic and informational purposes. In fact, for bearers where IP services are supported, ICMP is used as the WCMP.

15.5 WAP PROFILES AND CACHING

Since different WAP-enabled mobile devices have displays of different screen sizes and color depths, a WAP page might look different in different mobile devices, even the same version of WAP browser is used in the mobile devices. On the other hand, it

will be a waste of bandwidth if a color image is downloaded on a mobile device which can only display black and white images. Thus, it is clearly that if WAP contents are to be delivered to a mobile device, it is beneficial to know the capabilities of the mobile device first.

To describe the capabilities of a mobile device, WAP has defined a user agent profile (UAProf). The capabilities of a mobile device are related to software and hardware, including things like processor type, memory capacity, display size, browser type and version, network type, etc. The aim of using UAProf is to allow all elements of the WAP infrastructure (i.e., content servers, application servers, gateways, etc.) to provide mobile devices with device-specific contents.

A user agent profile is basically an XML document containing information about the hardware and software characteristics of a mobile device, and the network to which it will be connected. Therefore, it is not used to specify a user's preference. The user agent profile of a mobile device is stored in its manufacturer's server, called the profile repository. For example, the URL of the user agent profile for the Nokia 6230i cell phone is [333]:
http://nds1.nds.nokia.com/uaprof/N6230ir200.xml
An excerpt of the XML file is shown in Figure 15.8

```
.....
<prf:component>
  <rdf:Description rdf:ID="HardwarePlatform">
    <rdf:type rdf:resource="http://www.openmobilealliance.org/
      tech/profiles/UAPROF/ccppschema-20021212#HardwarePlatform"/>
    .....
    <prf:PixelAspectRatio>1x1</prf:PixelAspectRatio>
    <prf:ScreenSize>208x208</prf:ScreenSize>
    <prf:ScreenSizeChar>18x5</prf:ScreenSizeChar>
    <prf:StandardFontProportional>Yes</prf:StandardFontProportional>
    <prf:SoundOutputCapable>Yes</prf:SoundOutputCapable>
    <prf:TextInputCapable>Yes</prf:TextInputCapable>
    <prf:Vendor>Nokia</prf:Vendor>
    <prf:VoiceInputCapable>Yes</prf:VoiceInputCapable>
  </rdf:Description>
</prf:component>
.....
```

Figure 15.8 An excerpt of the user agent profile XML for the Nokia 6230i cell phone.

In order to provide mobile devices with device-specific contents, when a mobile device performs a request to a server, the URL of its user agent profile will be included in the header of the request message. A user agent profile contains the following components, each of which contains a number of different attributes:

Hardware Platform: It describes the hardware capabilities of the mobile device, such as color capability, screen size, and text input capability.

Software Platform: It describes the software capabilities or operating environment of the mobile device, such operating system type and version, character sets supported, and audio and video codecs supported.

Network Characteristics: It provides the information about the bearers supported, and their supported services, such as encryption methods supported.

Browser UA: It provides the information about the browser of the mobile device, e.g., browser name and version, and HTML version supported.

WAP Characteristics: It tells about the WAP features supported, such as WAP version and WML deck size.

Push Characteristics: It provides information about the push capabilities of the mobile device such as content types supported, and the maximum size of a push message.

MMS Characteristics: It provides information about the MMS (multimedia messaging service) capabilities, e.g., the maximum size of an MMS message, and the maximum image resolution.

Caching of frequently-accessed data items for answering subsequent requests is a promising technique in a wireless computing environment in order to reduce the response time. This is particularly useful for WAP-enabled mobile devices having limited capabilities. WAP has specified a caching model based on HTTP/1.1 caching as defined in RFC 2068 [118]. Like other adapted technologies like WMLScript and WTLS, in the WAP caching model a number of extensions and clarifications have been specified to facilitate the operation of HTTP/1.1 caching on mobile devices [244].

There are two main types of caching in the WAP architecture, namely caching by WAP devices and caching by WAP gateways. WAP-enabled mobile devices generally cache WML decks in order to reduce the time needed in reloading a page that has been previously loaded, without making a connection to the WAP gateway. This can lead to significant performance improvement in a high latency and low bandwidth wireless environment. However, for dynamic content, caching is inappropriate. The WAP caching model specifies the use of the Cache-control:must-revalidate header in order to signify whether a cached item should be revalidated when it becomes stale.

As for caching by WAP gateways, the WAP caching model specifies that a WAP gateway must faithfully implement the role of an HTTP/1.1 proxy with respect to caching and cache header transmission [244]. Specifically, a WAP gateway acts as a proxy server which caches documents requested by the network. Then the WAP gateway can fulfill subsequent requests from WAP clients, eliminating the round-trip time needed between the WAP gateway and the ordinary server.

15.6 PRACTICAL ILLUSTRATIONS—LOCATION-AWARE ADVERTISING SYSTEM

As mentioned earlier, the WAP push architecture has the nice feature that instead of sending a large push message, the push initiator can just send a short message (i.e., a service indicator) notifying the user of the push message. The short message comprises an embedded URL pointing to the actual content. This way, the user can choose to download the content later, or just simply discard the service indication. Hence, the WAP push architecture is useful for advertising. Suppose you are visiting a city that you are not familiar with and are wandering on the streets, then it will be convenient if there are some advertisements push to your mobile phone showing the information about the shops near to you or details of events in the city. However, one problem arises is how can the advertising system know your location information so as to trigger proper advertisements?

Aalto *et al.* [10] have proposed a permission-based location-aware mobile advertising system based on Bluetooth and WAP technologies. The system is called B-MAD (Bluetooth Mobile Advertising). Specifically, some locations (e.g., in shops) are installed with Bluetooth sensors, which periodically discover Bluetooth-enabled mobile devices and then trigger advertisements. Advertisements, on the other hand, are sent using the WAP push service.

Figure 15.9 shows the architecture of the B-MAD system. As can be seen, there are four elements in the system, namely the end user device, Bluetooth sensor, ad server, and the push sender (i.e., the push initiator in the WAP push architecture). The interaction of the four elements are described as follows:

User Registration: An end user subscribes to the mobile advertisement service through the ad server. During registration, the user needs to input his phone number (MSISDN) and the Bluetooth device address (BD_ADDR). The associated MSISDN and BD_ADDR is stored in a database of the ad server.

End User Bluetooth Device Scanning: The Bluetooth sensor periodically scans for nearby end user Bluetooth devices. If a Bluetooth device is discovered, the BD_ADDR of the device (and also the location identifier of the Bluetooth sensor) is sent to the ad server via a WAP connection.

Advertisement Preparation: When the ad server receives the request from the Bluetooth sensor, it checks in the database to see if there is a matched MSISDN for the received BD_ADDR. If there is a match, the ad server will prepare the required advertisement based on the location identifier of the Bluetooth sensor. Then, the ad server will pass the push sender the MSISDN of the recipient and descriptions of the advertisement (in this case, only short textual description of the advertisement and URL address pointing to the actual advertisement are sent).

WAP Push Message Sending: The push sender sends service indication message of the advertisement to the push proxy gateway using the push access protocol and XML over HTTP. The gateway then encodes the push message to a binary over the air (OTA) format, and uses a Short Message Service Center (SMSC) with SMS as the bearer to deliver the push message [10].

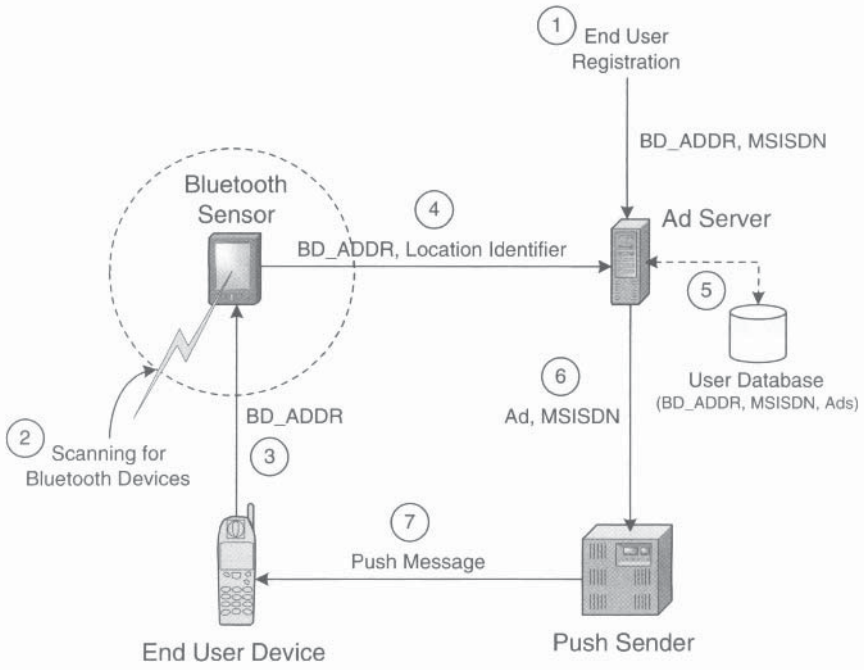


Figure 15.9 B-MAD permission-based location-aware mobile advertising system.

To evaluate the B-MAD system, Aalto *et al.* [10] have conducted quantitative evaluation in a laboratory environment and qualitative user evaluation in a field trial. They reported that there are inherent latencies in discovering end users' Bluetooth devices and delivering advertisements. This will be annoying to users if they receive the advertisements too late but they have already walked quite far away from the Bluetooth sensor. On the other hand, during user registration, the ad server should be modified to accept the user's preference on receiving advertisements so that only personalized advertisements are sent to the user.

15.7 SUMMARY

In this chapter, we have discussed how mobile users are supported with Internet information services through the WAP technology. Specifically, WAP addresses the various limitations of mobile devices, like limited processor power, memory capacity, battery life, input and output facilities. It also address the limitations of wireless networks, such as low bandwidth, high latency, unpredictable available and stability. The lightweight WAP protocol stack, WML, WMLScript, UAProf, and the WAP caching model are designed with the goals to address the limitations of mobile devices and wireless networks. Finally, we have presented how a mobile advertising system can be built with the WAP and Bluetooth technologies.

PROBLEMS

15.1 What is the use of UAProf? How can it enhance the efficiency of delivering WML content to mobile devices?

15.2 How does WAP address the problem of limited battery life of mobile devices?

15.3 Briefly describe the functions of a WAP gateway.

15.4 What are the advantages of using the WAP push service?

CHAPTER 16

TCP OVER WIRELESS

Let every man praise the bridge that carries him over.
—English Proverb, traced by Smith to 1678

16.1 INTRODUCTION

As we have seen, WAP is a specialized technology designed for adopting TCP/IP in a cellular network. However, the model underlying WAP is inappropriate in many other wireless environments (e.g., a satellite wireless link). Thus, there are many other solutions proposed for enabling TCP over wireless.

Interoperability:

In the past few years, we have witnessed that the Internet has extended its reach to the wireless networking environments. In fact, we are stepping into the ubiquitous mobile computing age. With the help of mobile devices like cell phones and personal digital assistants (PDAs), and 2.5G and 3G mobile data services like GPRS and UMTS, users are provided with ubiquitous mobile access to IP-based applications

Wireless Internet and Mobile Computing. By Yu-Kwong Ricky Kwok and Vincent K. N. Lau

Copyright © 2007 John Wiley & Sons, Inc.

in traditional wired networks. Thus, traditional Internet protocols are also used over wireless links. Since TCP/IP (Transmission Control Protocol/Internet Protocol) is the standard network protocol stack on the Internet, it is certain that TCP is used in different wireless environments. Not only does it leverage a large number of existing applications, but it also provides seamless integration with the wired Internet.

Performance:

The wireless users may commonly find that the provided services in wireless networks are not as good as wired networks because of the limited bandwidth and high bit error rate (BER) in wireless links. On the other hand, a straightforward migration of Internet protocols in traditional wired networks to wireless networks will result in poor performance. In particular, the TCP, which has many salient features that are useful in a wired network, needs significant modifications in order that it can deliver packets over the wireless links efficiently. Indeed, the congestion control mechanisms are particularly problematic in wireless environments. Due to the inherent reliability of wired networks, there is an assumption made by TCP that any packet loss is due to congestion. However, in wireless networks, packet losses are no longer mainly due to network congestion, but due to some wireless specific reasons such as random loss or collision in wireless networks.

Solving the problem of TCP performance degradation in wireless environments is very challenging because TCP is designed and tuned for wired networks. Unfortunately, compared with wired networks, wireless networks have many different characteristics, such as high bit-error-rate (BER), low channel bandwidth and occasional blackout. Recently, many adaptive TCP approaches for various wireless environments have been suggested. There are three main groups of approaches: link layer, split connection and end-to-end approaches. The major objective of these schemes is to make TCP respond more intelligently to the lossy wireless links.

16.2 TCP CONGESTION AND ERROR CONTROL

A Brief Introduction to TCP

Transmission Control Protocol (TCP) is the most commonly used transport layer protocol in Internet for providing reliable end-to-end data transfer. It specifies the procedures for connection setup and connection close, the usage of sequence numbers for reliable and in-order packet delivery, and the timeout retransmission mechanisms.

TCP is a connection oriented protocol and is designed specifically for use over an unreliable network. To establish a connection, a three-way handshake mechanism is adopted in TCP.

In TCP, reliability is achieved through the use of sequence number, cumulative ACKs, and retransmission techniques. Data packets are sent out in order, and each data packet or acknowledgment packet has a unique sequence number. If a packet is received out-of-order, the receiver acknowledges with the highest received in-order sequence number. Thus, each ACK indicates that all the packets whose sequence

numbers are less than the sequence number of the currently received ACK have been received. Therefore, ACKs for these out-of-order packets are called duplicate ACKs (DUPACKs). Obviously, TCP uses ACKs as a clocking mechanism to transmit data packets. In the case that no ACK is received during an interval called timeout, TCP uses retransmission to ensure the delivery of data packets.

To prevent sender from delivering too fast, TCP employs a flow control mechanism using advertisement window, which indicates the buffer size currently available at the receiver. The sender gets the advertised window size via the new coming ACK. Using the flow control mechanism, the sender is not allowed to transmit packets more than that the receiver can buffer; otherwise, excessive packets would be discarded.

The original TCP standard mentioned above does not specify any congestion control mechanisms and is not deployed in current Internet, but it is the basis of other TCP variants currently deployed in Internet.

TCP Reno

TCP Reno [165, 327] is widely deployed in Internet. It has a reliable window-based flow control and congestion control protocol. The sender utilizes ACKs to pace the transmission of segments, and interprets triple duplicate ACKs and retransmission timeout as congestion indication. TCP sender reduces the transmission rate by shrinking its window size when it detects a congestion. In Reno, four intertwined algorithms are included in the congestion control protocol. They are slow start, congestion avoidance, fast retransmit, and fast recovery. In addition, retransmission timeout is also employed in the congestion control mechanism.

Two variables are used in Reno, congestion window ($cwnd$) and slow start threshold ($ssth$). The variable $cwnd$ indicates the number of data packets which are allowed to be transmitted currently; $ssth$ is the threshold to determine whether the current congestion control of this connection is in the phase of slow start or congestion avoidance.

The main procedures in Reno's congestion control mechanism are explained in the following:

Slow Start (SS): SS is carried out at the beginning of a TCP connection or after a retransmission timeout. The purpose of SS is to prevent the sender from injecting multiple packets into the network. The SS mechanism prevents the sender from network congestion with burst data and gradually increases the amount of transmitting data to discover the available bandwidth. When a TCP connection is established, $cwnd$ is initialized to one packet. If packet losses are indicated by timeout, $cwnd$ is also set to one packet. Each time an ACK is arrived, $cwnd$ grows by one packet. With reference to Figure 16.1, the increase of $cwnd$ is exponential. The increase will end when $cwnd$ reaches $ssth$, which is given by:

$$ssth = \min\{cwnd, awnd\}, \tag{16.1}$$

where $cwnd$ is based on the estimation of perceived congestion by the sender, $awnd$ is advertised window imposed by the receiver and relative to the available

receiver buffer size in this connection. Therefore, SS can effectively avoid packet burst injecting into the network as well as quickly fill up the estimated Internet capacity.

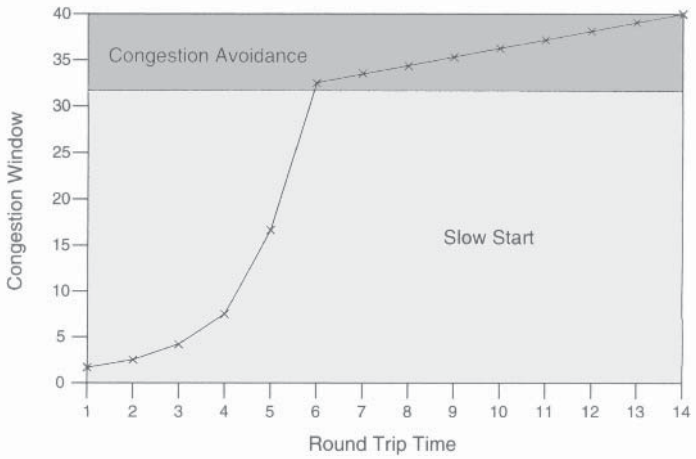


Figure 16.1 Variation of congestion window size due to slow start and congestion avoidance mechanisms.

Congestion Avoidance (CA): When $cwnd$ reaches the value of $ssth$ or receiving several DUPACKs, the congestion control enters the CA phase. In this phase, the TCP tries to slowly probe the network for extra bandwidth. $cwnd$ is increased by one packet per RTT (round trip time). Each time an ACK is arrived, $cwnd$ is increased by:

$$\Delta cwnd = \frac{packet\_size \times packet\_size}{cwnd}, \tag{16.2}$$

where the packet size and $cwnd$ are maintained in bytes.

Compared with SS, CA leads to linear phase of $cwnd$ growth (see Figure 16.1). In fact, the increase of $cwnd$ is no more than one packet in every RTT.

No matter whether the congestion control is in SS or CA, when packet losses are detected, $ssth$ must be set to half of its original value.

Fast Retransmit and Fast Recovery: When an out-of-order packet reaches the receiver, a DUPACK is sent back to the sender. The purpose of DUPACK is to let the sender know that an out-of-order packet is received and what sequence number is expected, in addition to maintaining ACK clock. However, the TCP sender does not know which packet triggers this DUPACK, and whether the expected packet is lost or out-or-order. Then it waits for a small number

of DUPACKs to be received. When the Kth DUPACK is received, it strongly indicates the expected packet has been lost. Then TCP performs retransmission without waiting for timeout. Typically, K is equal to 3. This mechanism is called Fast Retransmission.

Generally, fast retransmit and fast recovery are implemented together. After fast retransmit is triggered, besides the sender retransmits the lost packet, fast recovery algorithm is also employed to govern the transmission until a non-duplicate ACK arrives. Since DUPACK is generated by receiving new and out-of-order packets, there is still data flowing between sender and receiver. Fast recovery is used to avoid reducing the flow dramatically and to maintain flow rate by continuing to send packets according to CA algorithm. When receiving three DUPACKs, the sender retransmits the expected packet, $ssth$ is halved and the $cwnd$ is set to $ssth$ plus three. When another DUPACK is received, the sender transmits one packet and $cwnd$ is increased by one if $cwnd$ is less than $awnd$. After TCP finishes fast recovery, it enters CA with linearly increasing $cwnd$.

Retransmission Timeout: Since TCP is ACK-clocked, round trip time (RTT) which is the time of data packet travels from the sender to receiver plus ACK travels from the receiver to the sender is an important variable to predict packet loss. The value of estimated RTT is applied to another variable called retransmit timeout (RTO), which is the maximum time to wait for an ACK after sending a data packet. RTO governs if retransmission is needed. The estimation of RTO is significant, because if RTO is too small, unnecessary retransmission is triggered; if RTO is too large, the sender will wait without retransmission until the timer expires. The value of RTO is based on the RTT as governed by:

$$S\_RTT_{i+1} = \alpha \times M\_RTT_i + (1 - \alpha) \times S\_RTT_i, \tag{16.3}$$

where S_RTT is the smoothed RTT, M_RTT is the most recently measured RTT, α is set to 1/8 [164].

Thus,

$$RTO = S\_RTT + 4 \times \Delta RTT, \tag{16.4}$$

where ΔRTT is the estimate of standard deviation of the difference between S_RTT and M_RTT.

Retransmission timeout mechanism is a supplement when no ACK clock is received for a long time. After the retransmission timer is expired, the value of RTO is doubled. This mechanism is called RTO backoff [183], which can weaken congestion in the network and help the connection leave the congested state.

16.3 DEFICIENCIES AND OVERVIEW OF SOLUTIONS

TCP has been tuned to perform well in traditional networks comprising wired links and stationary hosts. The basic assumption is that congestion is the primary reason for packet losses in wired networks. When packet losses are detected, the TCP sender exponentially throttles down the congestion window size before retransmitting lost packets. It then backs off its retransmission timer, and enters the congestion avoidance phase. All these mechanisms are designed to reduce the load injected into the network. These mechanisms work well in a wired network where the bit error rate (BER) is typically low to allow any packet loss to be treated as an indication of congestion.

However, the situation of wireless networks is much more complicated than that in wired networks. Only congestion control is not enough to deal with so many different sources of TCP performance degradation in wireless networks. We classify the reasons for the deficiencies into five groups. They are high bit error rate, high round trip time, occasional disconnection, the MAC layer and asymmetric bandwidth:

High Bit Error Rate: In particular, high bit error rate is an important characteristic in wireless network. It occurs for different reasons such as path loss, fading, noise and interference.

High Round Trip Time: The high round trip time may reduce the increase rate of congestion window, even in the case of loss of ACK. Since congestion window is a function of number of ACKs, the slow start phase, which is critical time to occupy precious channel, increase slowly.

Occasional Disconnection: Occasional blackout is due to the device mobility. As a mobile host travels between wireless cells, the task of forward data between the mobile host and wired network must be transferred to the new cell's base station. This is called handoff. A long handoff time may result in long latency and packet losses. In ad hoc wireless networks, network partitioning occurs when a give mobile node moves away or is interrupted by the medium, then the neighboring nodes are separated into two parts of the network. Also, the adverse effects of the partitioning on TCP depends on its duration. A long partitioning time may lead to connection abortion.

The MAC Layer: IEEE 802.11 is standard MAC layer protocol for wireless LANs. It can support some ad hoc network architectures. However, this protocol is not designed for such multihop networks. It does not perform well in multihop ad hoc networks [332, 354]. Hidden terminal problem may lead to data collision and exposed terminal problem may result in under-utilization of the available bandwidth (as detailed in Chapter 9.4. And serious unfairness occurs when TCP connections cooperate with IEEE 802.11. Simulations show the TCP performance decreases sharply with the increase of the number of hop counts.

Asymmetric Bandwidth: Asymmetric bandwidth means low bandwidth for the ACK transmission. In some wireless networks, low bandwidth is used for the ACK traverse. Even the ACKs are smaller in size than data packets, the bandwidth of reverse channel cannot afford high rate of ACKs. The result is congestion and loss in ACK channel.

The main reason for the TCP performance degradation in wireless networks is the reaction to non-congestion losses. Generally, if the TCP sender could distinguish different losses, or non-congestion losses are shielded from the sender, the unnecessary congestion control could be avoided. In this context, two methods are used to enhance the TCP performance. The first one is to hide the non-congestion losses from the TCP sender. In this method, the non-congestion losses could be solved locally and thus, most of the losses seen by the TCP sender are indeed caused by congestion. The second on is TCP aware protocol. Using this method, the TCP sender should respond to different losses by different actions.

The existing schemes for enhancing TCP performance in wireless networks can be classified into three groups: link layer, split connection and end-to-end approaches. Simply put, these approaches target networks where only the last hop communicates through a wireless link, as illustrated in Figure 16.2. In this kind of wired-cum-wireless networks, the mobile host communicates with the base station which is connected with a wired backbone. This scenario is common in daily life, such as Internet access through a GPRS phone.

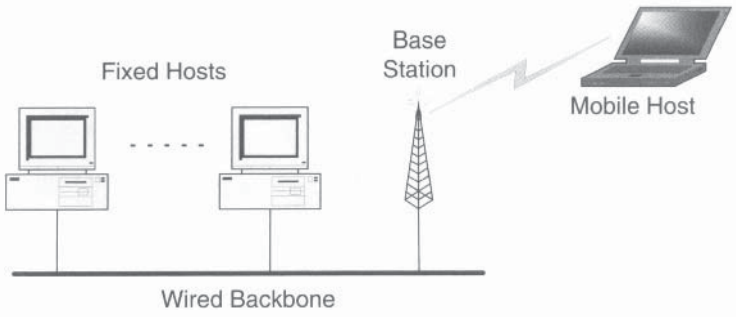


Figure 16.2 Last hop wireless link scenario.

16.4 LINK LAYER APPROACHES

The link layer is governed by unreliable, connectionless and best-effort delivery mechanisms. If the link layer could be reliable and correct the packet errors, TCP sender has less possibility to act with losses. Therefore, the purpose of link layer approaches is to improve the reliability of the wireless network. They attempt to hide the link-related

losses from the TCP sender using local remedies, such as local retransmissions, in the wireless link.

Traditional Link Layer Schemes

Traditional approaches for loss recovery at the link layer are FEC (forward error correction) and ARQ (automatic repeat request). The former is to let the receiver recover loss promptly using the redundant correction bits, at the expense of a lower effective throughput even in the case of an error-free transmission. The latter is to let the sender repeat the transmission of any lost packet. These two schemes are generally used in the lossy environments. Many proposed algorithms [19, 25, 253] are based on combined FEC and ARQ.

Asymmetric Reliable Mobile Access In Link-layer (AIRMAIL) [19] is an asymmetric protocol because the base station is obliged for making decisions whenever it transmits or receives packets. The base station sends periodic status messages to allow that the mobile host combines several acknowledgments into a single one. It employs a combination of ARQ and FEC for loss recovery to provide reliable links. It modified the ARQ retransmit strategy: the packets of the whole window are transmitted before a link layer acknowledgment is received by the link layer sender. Furthermore, an adaptive algorithm is employed based on three levels of FEC and the error correction only can be implemented after the whole window. This may cause TCP timeout for the long time latency. AIRMAIL also has mechanism to deal with handoff. Figure 16.3 shows how a mobile terminal (MT) works when handoff occurs using AIRMAIL. It should be noted that AIRMAIL allows the mobile transmitter to continue transmitting data while the handoff is occurring.

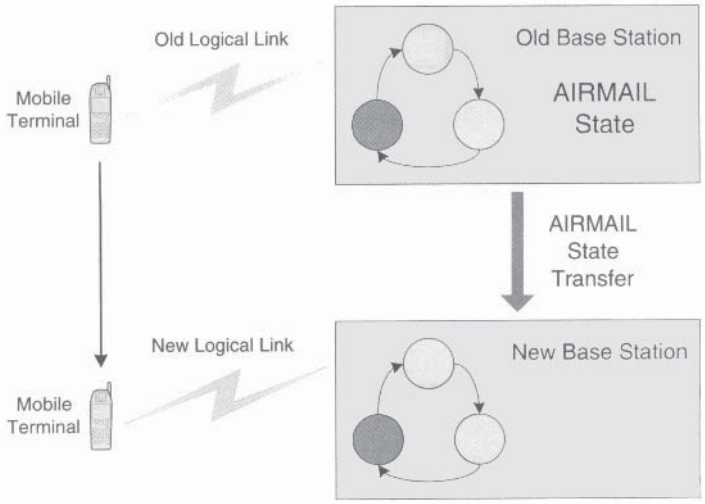


Figure 16.3 Mobility management in AIRMAIL [19].

Radio Link Protocol (RLP) [253] is a link layer based approach designed for local recovery of packet loss by a judicious use of the sequence numbers. It is based on a point-to-point ARQ for radio channels. In this protocol, a packet is retransmitted only if the transmitter is sure that it is lost. This is implemented by using the feedback packets (see Figure 16.4) from the receiver together with the sequence number of packet and a send sequence number at the sender. Thus, it is efficient to make sure the receiver only gets one copy of each packet. An enhanced version of RLP, which piggybacks partial receiver state in the reverse channel on uses data packets, avoids to be idle when all retransmissions are completed.

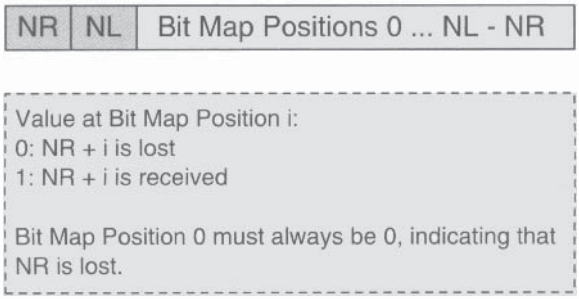


Figure 16.4 Feedback packet [253].

Snoop and Its Enhancements

Snoop [22] is a link layer aware TCP scheme. It is based on a special module called the Snoop agent at the base station (see Figure 16.5). The Snoop agent monitors every packet that travels through the wireless TCP connection in both directions and processes them with different measures according to the types of packets. The most important function of the Snoop agent is to recover the lost packet and retransmit them. In the Snoop protocol, it is assumed that all packet losses in wireless links are not due to congestion. As such, the Snoop agent maintains a cache of TCP packets sent across the link that have not yet been acknowledged by the mobile receiver. When packet loss is detected by either three duplicated ACKs or local time-out, the Snoop agent retransmits the packet if it has been cached and suppresses the duplicate acknowledgments. The fixed TCP sender will not trigger the congestion control or avoidance until time-out. The merit of the Snoop protocol is to retransmit losses locally and suppress the duplicate acknowledgments, thereby avoiding unnecessary fast retransmit and congestion control at the fixed sender.

NewSnoop [155] and FDA [154] are enhancements of Snoop. NewSnoop uses the hierarchical structure to enhance the performance during the handoff. It adds a tag in the packet at base station agent to identify the types of cached packets. Moreover, the Mobile Switch Center also caches the unacknowledged packets to help minimize the latency of retransmission during handoff. Snoop and NewSnoop are efficient when a single link is used to connect the mobile device to the base station. When

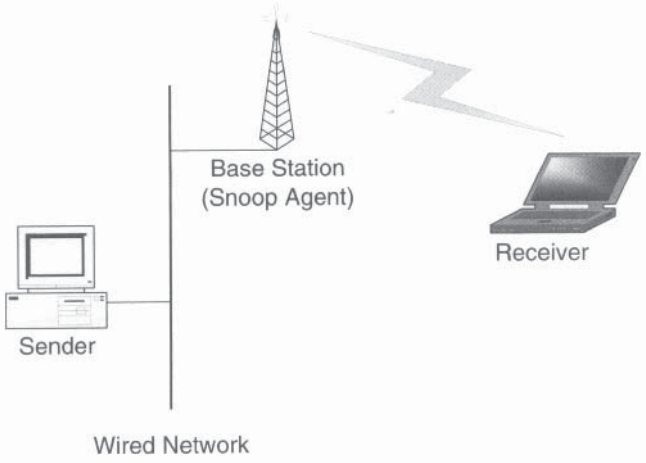


Figure 16.5 Snoop agent at the base station.

multiple connections are used simultaneously, the base station cannot buffer so many unacknowledged packets. FDA is proposed to solve this problem. A predefined buffer threshold indicates the incipient congestion. After the detection of incipient congestion, a special congestion notification, called three forced duplicated ACKs, is generated by the base station instead of the mobile host. This mechanism can potentially reduce the congestion response time for the fixed host. On the other hand, quality guaranteed cache release functions designed to minimize the change of multiple packet dropping at the base station during congestion.

16.5 SPLIT CONNECTION APPROACHES

Split-connection TCP schemes split the logical connection into two parts using the base-station as the break-point—one connection between the sender and the base station, the other between the base station and the receiver. The main idea behind the split connection approaches is to isolate the mobility and wireless related problems from the existing protocols in order to hide the wireless packet loses from the fixed TCP sender. Consequently, the performance of the TCP connection over the fixed part is not influenced by the wireless link and the wired connection does not need any change in existing software on the fixed hosts. The protocol used in the wireless link can be TCP or some other specialized protocols for wireless. Example protocols are I-TCP [20], M-TCP [42], and MTCP [360].

Indirect-TCP (I-TCP) [20] is a split connection approach using TCP for its connection over the wireless link. It attempts to separate loss over the wireless link from what over the wired links and hide packet loss from the TCP sender (see Figure 16.6).

Two components are included in I-TCP: one is on the mobile host and the other on the base station. The component on the base station is used to pump data from one part to other. The other component is similar in functionality and interface to the socket and make the communication transparent to the mobile host. But the TCP resulting does not perform well in wireless links. Furthermore, it violates the end-to-end semantics since the ACKs can reach the fixed sender even before the packets actually reach the mobile device. Therefore, it is not easily deployed in the Internet.

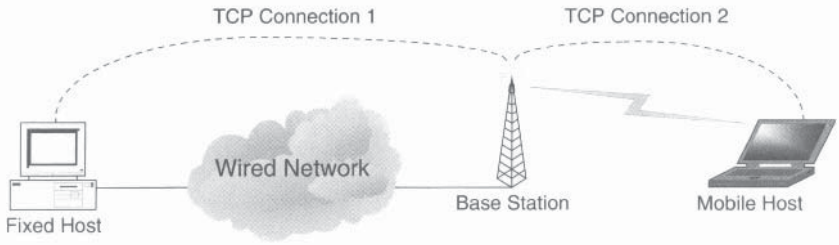


Figure 16.6 Splitting a TCP connection into two connections.

MTCP [360] is based on a very similar idea as in I-TCP. It protects the connection over the wired network from the impact of the short disconnection over the wireless link. The authors [360] proposes a session layer protocol called mobile host protocol (MHP) at the base station and the mobile host in order to compensate for the unreliability of the wireless link. Two implementations are suggested for the session layer. The first one uses TCP and the other uses selective repeat protocol (SRP), which is designed to recover the packet loss quickly over the wireless link.

Another scheme, called M-TCP [42], also splits the TCP connection but preserves the end-toend TCP semantics. This scheme is targeted for the frequent disconnection and low bandwidth. The TCP connection is also split at the base station. The three-layer hierarchical architecture is set up to avoid handoff. As shown in Figure 16.7, the mobile hosts are in the lowest layer, the base stations are in the second layer, and the supervisor hosts connected to the wired networks are at the top level of the hierarchy. The M-TCP protocol is mainly used at the wireless link. The base station adds two TCP clients: one for wired connection; one for wireless connection. When packets or ACKs reach the base station, they are delivered from one client to another. Then the packets are forwarded to the other end point. And a fixed bandwidths are dynamically assigned to the mobile hosts to reduce power consumption.

Another type of splitting TCP connection is called the Remote Socket Architecture (ReSoA) [314]. The main idea is to merge the TCP/IP stacks of mobile host into the base station and leave only user interface on mobile host as depicted in Figure 16.8. The connection between mobile host and base station is implemented by last hop protocol (LHP) which is a reliable protocol. In this way, base station acts as a proxy for the wireless end points.

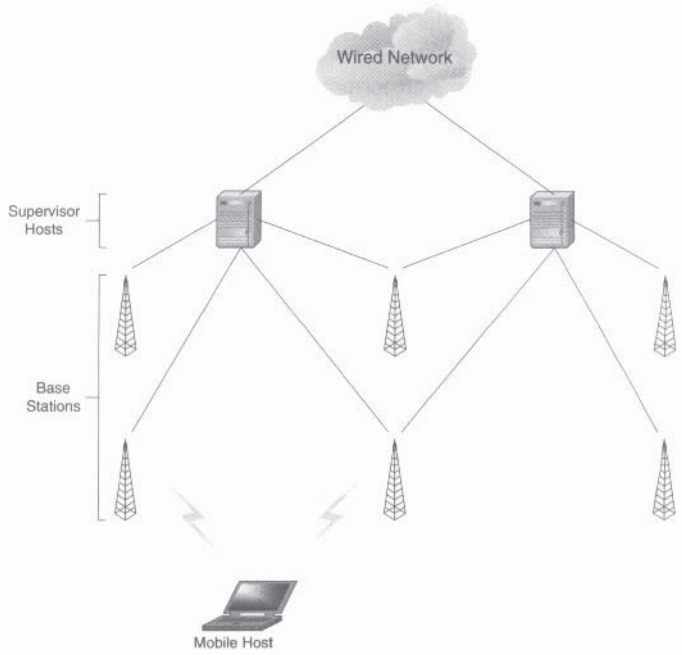


Figure 16.7 Three-level architecture of M-TCP.

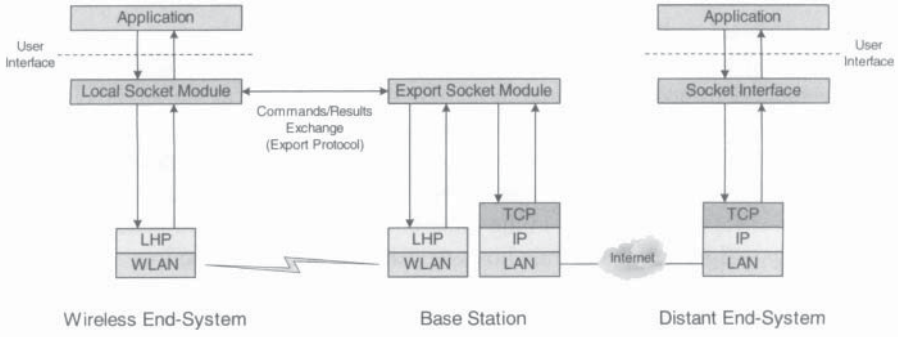


Figure 16.8 Wireless Internet access using ReSoA [314].

However, split-connection schemes must maintain the state of each TCP connection at the base station. As a result, a heavy overhead is incurred, resulting in a low efficiency.

16.6 END-TO-END APPROACHES

End-to-end approaches are designed based on the basic TCP protocol (i.e., Tahoe [164]) to handle packet losses. The first group is TCP Reno's variants, including NewReno, SACK and so on. The second group is based on the help of intermediate routes to predict the congestion and then inform the sender explicitly.

TCP Reno's Variants

Different loss recovery mechanisms are used in the TCP sender to improve the performance. Specifically, the TCP sender uses the cumulative ACKs to determine if packets have reached the receiver and provide reliability by retransmitting lost packets. Packet losses are detected by either three duplicate ACKs or absence of an ACK in a time-out interval. TCP-Tahoe [164] uses the fast retransmit method to recover any loss and enter slow start phase in which it increases the window exponentially.

Fast Recovery is implemented on top of Fast Retransmit in TCP-Reno [165, 327]. Fast Recovery can effectively estimate the amount of packets in flight while recovering from losses. The TCP sender could continue transmitting new packets if this number is below the estimation of the network capacity. The purpose is to preserve the ACK self-clock and avoid the time-out. Then, the congestion window size does not need to slow down dramatically to enter slow start phase, which will result in unnecessary throughput reduction. Instead, it stays in the congestion avoidance phase, where it increases window linearly.

In TCP-Reno, the Fast Recovery algorithm is conservative. Only the first loss packet in the congestion window can be recovered. This is because when an ACK of the first loss packet is received, it leaves Fast recovery phase. This prevents the sender from detecting the other losses through Fast Retransmit. So the other losses in the same congestion window cannot be found until after the long time-out. TCP-NewReno [122, 150] solves this problem by introducing partial ACK—the ACK which covers new data but not all the data outstanding when loss was detected. The objective of doing this is to detect multiple losses in the same congestion window. The TCP sender then does not wait for the duplicate ACK, but retransmits the loss packet immediately when partial ACK is received. The Fast Recovery will end until all packet losses are detected in the same congestion window.

Cumulative ACKs report the number of the packets that have been accumulated. But they do not give the sender much useful information about successfully received packets. If an ACK can carry information about the recently received packets, loss recovery mechanism can be started once receiving such an ACK. SACK (Selective Acknowledgments) [124, 236] are designed to provide such information. They are added as an option to TCP. This option can at most provide three non-contiguous

blocks of data which have been received successfully by the receiver. Each block of data is described by its starting and ending sequence number. The earlier is the block, the later have the packets been received. That is, the SACK option always reports the most recently received packets. The redundant appearance of several SACKs can provide enough information about packet losses.

SMART [188] contains not only cumulative ACKs but also the sequence number of the packet which caused the receiver to generate the ACK. The key idea in this algorithm is to build the bitmask of correctly received packets at the sender, instead of carrying it in the ACK header. If the sender detects any gap in the bit-mask, it assumes that the missing packets have been lost and then retransmits the lost packets. However, the out-of-order packets would be misunderstood to be loss.

Explicit Notification

Traditionally, route queue length is set to be a maximum length for each queue. All incoming packets are queued in the router until reaching maximum length. After that, new incoming packets are dropped till old packets are sent out and queue length decreases. This is the well-known "drop tail", because the packet that arrived most recently is dropped when the queue is full. Reference [40] points out two drawbacks of this drop tail technique. The first one is lockout. In some situations, a single or a few connections can monopolize the queue size. The other drawback is full queues. Allowing full queues could result in maintaining queue full for a long time. Based on the above reasons, active queue management is needed. Random early drop (RED) [40, 123] is an active queue management algorithm, which detects incipient congestion by computing the average queue size (avg). RED drops packets depending on the relationship between two thresholds and the average queue size. The avg is a time-averaged length, not an instantaneous one. Besides RED, BLUE [117] is also an active queue management scheme. It aims to remedy the shortcomings of RED.

Floyd and Ramakrishnan [121, 292] first suggested the explicit congestion notification (ECN) mechanism based on RED queue management algorithm. The objective of ECN is to reduce packet drop in the router in order to avoid unnecessary congestion control mechanisms at the TCP sender. Specifically, when a router experience incipient congestion, one randomly selected packet in the queue is marked. The receiver accordingly sends a special ACK packet upon reception of the marked packet. The sender will then reduces the congestion window (which would have been increase) upon reception of the special ACK.

Liu and Jain [218] proposed a strategy called mark front, which can lead to a fast congestion feedback. In the mark front algorithm, a router will mark the packet, which is at the head of the queue, when incipient congestion is detected. Because such a packet is to be sent out first, the feedback time of the mark front strategy is considerably shorter than that of the mark tail method.

Hammann and Walrand [142] suggested an algorithm called New-ECN, which works by preventing a fast connection from opening its congestion window too quickly, but instead enabling a slow connection to open its window more aggressively.

When the marked packet is echoed to the sender, not only the congestion window size is decreased but also the rate of the congestion window increase is modified.

Peng *et al.* [275] proposed the W-ECN algorithm for the wireless environment. In this algorithm, upon noting the first lost packet due to buffer overflow in the router, the headers of passing packets in the queue are marked. In this manner, the sender knows that a packet loss happened due to network congestion but not because of random loss due to degraded channel quality.

Floyd proposed explicit loss notification (ELN) [120]. The explicit information is transmitted to the sender. And then the sender reacts by retransmitting the lost packet without reducing the congestion window. The most difficult task in ELN is to get the information about loss because the corrupted packets have been discarded before they reach TCP.

Bakshi *et al.* proposed explicit bad state notification (EBSN) [21]. Although local recovery by the base station could resolve the packet drop in the wireless link, timeout could happen and then trigger redundant retransmitted packets. The main objective of EBSN is to eliminate the timeout at the fixed host when base station is in the local error recovery. When the wireless link is in the bad state, the base station uses local recovery and the notification is sent to the fixed host to reset the timeout at the fixed host.

16.7 PRACTICAL ILLUSTRATIONS—WIRELESS WAN

On improving the TCP performance over wireless, most research is focused on mathematical analysis and simulations. In [56], Chakravorty *et al.* introduced a detailed experiment-based evaluation and comparison of optimization techniques in a commercial wireless wide-area networks (WWANs). They focused on the performance of web browsing applications in WWAN cellular environments. In their study, they implemented various optimization techniques across different layers (i.e., application, session, transport and link layers) of the protocol stack to investigate the cross-layer interactions of those techniques and their impact on the application performance. The key observation of their work is that even though TCP is relatively well-tuned to perform well in WWAN cellular environments, the performance of HTTP applications is not satisfactory because of the severe mismatch between TCP and default HTTP protocols. In addition, with suitable optimizations implemented in the application and session layers and the use of proxies, the end-user web-browsing experience can be enhanced by at least a factor of two.

Since the focus of this chapter is wireless TCP, the work of Chakravorty *et al.* [56] about performance optimization techniques in the transport layer will be discussed. For optimization techniques in other layers, interested reader is referred to [56].

The experimental testbed is shown in Figure 16.9, which is a GPRS-based WWAN network. Two nodes, SGSN (Serving GPRS Support Node) and GGSN (Gateway GPRS Service Node), are added to the GSM network to support GPRS. SGSN per-

forms functions like signaling, cell selection, routing and handoff between different BSCs (Base Switching Centers). GGSN is the gateway between GPRS network and fixed IP routing network.

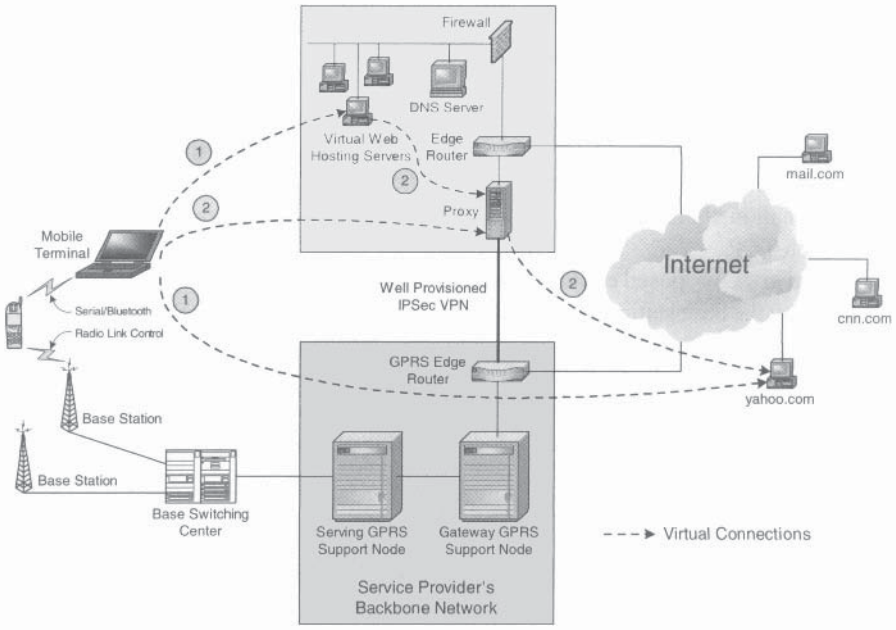


Figure 16.9 WWAN experimental testbed [56].

The network setup is as follows. A mobile terminal (MT) such as a laptop connects to the WWAN network via a GPRS phone. To use the WWAN network, the MT first attaches itself to the GGSN through a signaling procedure and establishes a point-to-point protocol (PPP) connection with the GGSN. Then, the MT successfully connects with the GPRS network, and is dynamically assigned an IP address. Finally, when the MT moves through the network, the WWAN network switches its data to the assigned IP address.

Since the contents of popular websites change frequently, different download attempts may show significant differences in the download content structure, and hence different response time. As such, the comparison of web download from different experiments in the relatively low-bandwidth WWAN network could be meaningless. In order to avoid this problem, in the WWAN experiments conducted in [56], a virtual web hosting system is implemented in a laboratory where the contents of the popular websites are replicated into a set of web servers with public domain names. This way, the MT accesses the virtually hosted web pages using WWAN networks as if they were accessed from actual servers, with the nice property that the access results can be repeated and reproduced.

As mentioned before, the main objective of the WWAN experiments is to evaluate the web-browsing performance over a WWAN network. The MT downloads web contents over the WWAN link via two different paths. The first one is to directly download from the real web servers or virtually hosted seb-servers. This case is shown by Label 1 in Figure 16.9. The second one is to download through a proxy, which is connected to the cellular provider's network via a well provisioned IPSec VPN. This case is shown by Label 2 in Figure 16.9.

In [56], the authors implemented a performance optimization scheme called TCP-WWAN [57] in the transport layer to study the impact of the transport layer to the web-browsing performance. TCP-WWAN is a transparent-proxy solution and uses a 'transparent' proxy located in the cellular provider's network. It has two characteristics in the WWAN network. Firstly, it uses a pre-determined value of bandwidth-delay product to replace the TCP slow start. In addition, it performs aggressive recovery during packet loss and link stalls. Such kind of aggressive behavior makes TCP-WWAN only implemented in the cellular network where appropriate bandwidth sharing mechanism are implemented. Secondly, TCP-WWAN estimates available bandwidth on the wireless link so as to regulate the flow of TCP packets to the MT. Thus, TCP-WWAN can reduce the queue size and prevent spurious timeout, which on the other hand avoid the occurrence of wireless link stalls.

The evaluation of TCP-WWAN was performed using Mozilla browser version 1.4. Instead of using the default setting (HTTP:/1.1, two persistent TCP connections), the browser was modified to use six persistent TCP connections. A 1.4 GHz laptop with Linux installed runs the Mozilla browser, and connects with WWAN GPRS network by using a '3 + 1' GPRS phone. The WWAN operates in the 1.8–1.9 GHz band and uses CS-2 FEC coding scheme. In all experiments, the MT was kept stationary to avoid link variations.

Experiment results show that TCP-WWAN can achieve between 5–13% additional benefit in the browsing of various popular websites. This performance improvement is significant for web-browsing over WWAN links. However, as discussed in [56], the performance improvement can be even more significant when optimizations are done in the application and session layers (about 48–61% improvements), which is not the scope of this chapter.

16.8 SUMMARY

In this chapter, we have discussed the reasons for TCP performance degradation in wireless networks. Particularly, we talk about the reason why the congestion control mechanism can harm the TCP performance significantly in wireless environments. Then, we have introduced a number of representative adaptive TCP algorithms in different wireless environments. These schemes are classified into three groups: link layer, split connection and end-to-end. Finally, as a practical illustration, we have

talked about a study of enhancing TCP performance for web-browsing in a commercial wireless wide-area network (WWAN) testbed.

PROBLEMS

16.1 What are the reasons which lead to the TCP performance degradation in wireless networks?

16.2 What are the design objectives of link layer approaches on improving TCP performance in wireless environments?

16.3 For web-browsing in an environment like wireless wide-area network (WWAN), if a web page consists of numerous HTTP objects of small sizes and TCP Reno is adopted, how will be the web-browsing performance? And why?

PART V

WIRELESS RESOURCES MANAGEMENT

This Page Intentionally Left Blank

CHAPTER 17

WIRELESS PACKET SCHEDULING

Fair as the moon.

—Solomon's Song VI, 10

17.1 INTRODUCTION

In many situations, the wireless medium is a resource shared among all wireless devices (e.g., in a WLAN). Thus, allocation of this shared resource to the users is an important component in the system. Specifically, such a scheduling mechanism is needed for both the uplink (from the devices to the base station or access point) and the downlink (from the base station or access point to the devices) directions.

Interoperability:

Existing wireless packet scheduling algorithms can be classified according to the fairness notions they employ. The key difference is the scheduling set induced by the channel model used. In algorithms with greedy fairness notions, the scheduling set includes only the backlogged flows with "good" channel state. In algorithms with

Wireless Internet and Mobile Computing. By Yu-Kwong Ricky Kwok and Vincent K. N. Lau
Copyright © 2007 John Wiley & Sons, Inc.

realistic fairness notions, it includes all the backlogged flows except the ones with the worst channel state (i.e., cannot transmit any packet). Greedy fairness notions simplify the wireless packet scheduling problem in that the scheduling set is homogeneous with respect to channel state. However, greedy fairness notions are unfair to the flows with channel states between the worst and the best. On the other hand, when realistic fairness notions are used, the scheduling set is larger, but the channel states of the flows in the scheduling set are not the same. This makes the design of fairness notions more complicated.

Performance:

Performance of a scheduler can be characterized by a multitude of metrics. The most widely used metrics are throughput (rate) and fairness. Throughput is obviously important in supporting users' QoS requirements. For instance, many applications required a minimum level of throughput support in order to be viable. By contrast, fairness is a much more subtle issue. First of all, it is difficult to define a universally accepted fairness metric. Indeed, there are many different fairness metrics suggested in the literature, such as proportional fairness, max-min fairness, etc. Secondly, as we will see, fairness is inherently in conflict with traditional metrics such as throughput. Specifically, in order to support a certain level of fairness, the overall system throughput might be reduced. Designing an integrated metric combining throughput and fairness is still a hot research topic. Other useful metrics for evaluating a scheduler include delay, overhead, efficiency, etc.

17.2 THE SCHEDULING PROBLEM

Packet scheduling algorithms have been extensively studied in wired networks. In a packet-switching network, packets coming from different connections arrive at the switching nodes (i.e., the routers) and wait to be delivered. Since each switching node can only deliver a limited number of packets at a time, a certain packet scheduling algorithm is needed at the switching node to decide the order in which the packets are served. On making the scheduling decision, the switching node may need to maintain some QoS guarantees (e.g., bandwidth, delay, fairness) for different connections from different end hosts.

Similar to the scenario in wired networks, a scheduling algorithm is needed at the base station to decide which mobile terminals will be served. For example, in a cellular phone or wireless LAN network, typically there is a central machine (i.e., the base station or the access point), which serves the mobile terminals with heterogeneous traffic demands (e.g., media stream data such as video or voice, and file data). To minimize collisions and delays of data transmissions of the mobile terminals, some coordination schemes, known as MAC (multiple access control) protocols, are employed. A scheduling policy is usually implicitly or explicitly incorporated in a MAC protocol. The goal of the scheduling algorithm is to judiciously handle the MAC problem for the shared wireless communication medium (specifically the uplink—the

link from the mobile terminals to the base station/access point, which is the major resource contended by the users) such that the utility of the medium is maximized while satisfying users' demands.

However, scheduling algorithms developed for the wired networks are not applicable to the wireless environment because of the inherently different characteristics of the wireless channel.

The Wireless Channel:

In contrast to the transmission quality of a link in a wired network, which is stationary after it is set, the wireless channel condition is continually affected by short-term fading, long-term shadowing, and depends on the mobility of mobile terminals, surrounding obstacles like buildings, mountains, etc. As these effects are location-dependent and time-varying, the signal strength of the wireless channel can vary widely in amplitude and phase. Thus, only a subset of the waiting flows (commonly called the backlogged flows), which is experiencing good channel conditions, is eligible for consideration by the scheduler. Moreover, this subset is time-varying. As illustrated in Figure 17.1, MT (mobile terminal) A is having a comparatively poorer channel state because it is far away from the base station (BS) and it encounters deep fading. On the other hand, MT B may have a better channel state as it is much nearer to the BS. MT D may have very poor channel state as its signal is obstructed by a large building. Thus, it may turn out that only MTs A, B and C can be scheduled, and they have different channel states.

To simplify the scheduling problem significantly, a two-state Markov chain channel model is commonly used [33, 52, 227]. Under this channel model, the wireless channel has just two states—either "good" or "bad". In a "good" channel state, a flow can transmit using full bandwidth, while in a "bad" channel state, a flow cannot transmit any packet. The model is depicted in Figure 17.2 As shown in the figure, P_g is the probability that the channel state in the next time slot is "good" given that it is "bad" in the current time slot. Likewise, P_e is the probability that the channel state in the next time slot is "bad" given that it is "good" in the current time slot.

The major drawback of the two-state Markov chain channel model is that it is unrealistic because packets can still be transmitted even if the channel is not in the perfect state (the "good" state). In reality, using some channel-adaptive technique like ABICM (Adaptive Bit-by-Bit Interleaved Channel Modulation) [198], the transmitter/receiver in a wireless network can exploit the time-varying nature of the channel and accordingly adjust the effective throughput. Specifically, when the channel state is not good due to multipath fading or shadowing (channel state can be determined by checking the pilot symbols in a feedback channel or reverse link), the amount of protection can be readjusted by choosing a different channel coding and modulation mode [198]. Thus, even in a so-called "bad' channel state, an MT can in fact transmit data and realize a possibly lower effective throughput, instead of being totally unable to transmit.

Fairness:

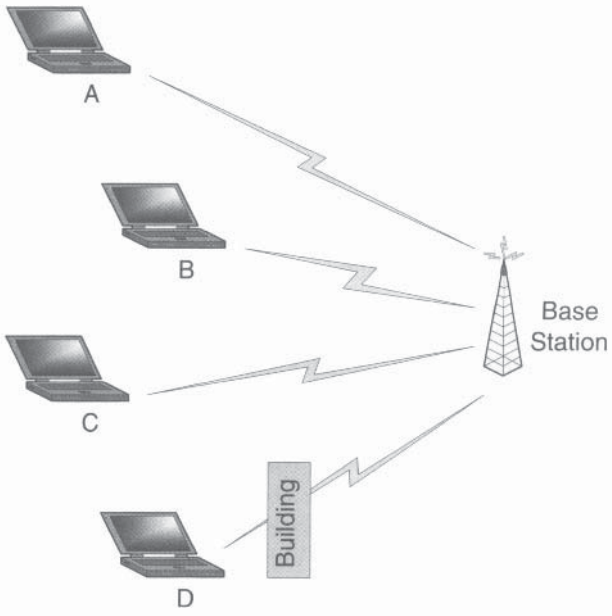


Figure 17.1 A wireless packet scheduling scenario.

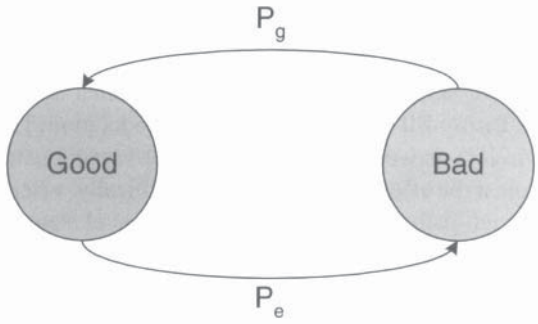


Figure 17.2 The two-state Markov chain channel model.

In a wired network, maintaining fairness and optimizing overall throughput (i.e., optimizing bandwidth utilization) are not conflicting goals. However, in a wireless network, if optimizing throughput is the only principal goal, an ideal method is to schedule among the backlogged flows with the decreasing order of their channel states. For example, consider two backlogged flows i and j, where i belongs to the set of flows with best channel state and j belongs to the set of flows with channel states between the best and worst. If the scheduler always gives the highest priority to the flow with the best channel state, flow i will be served continuously while flow j will be starved. This scheduling philosophy can maximize the overall throughput, but it is obviously unfair.

On the other hand, if fairness is the dominating goal such that it must be maintained at any time instant, an MT should be scheduled to transmit even when it encounters a very bad channel state. In the previous example, flow j will also be scheduled to transmit. Inevitably, such a policy degrades the overall effective throughput since it schedules the flows without the best channel state to transmit. Indeed, if we employ the classical fairness notions in the wired networks (defined with respect to resources only, without regard to utilization) and try to guarantee the same effective throughput between the two flows, and suppose that a flow j happens to have a very bad channel state, then the system will give much more time slots to flow j in order to let it have the same effective throughput as achieved by flow i. It is unfair to flow i because the system allocates much more time slots to flow j with such a poorer channel state. If those time slots were allocated to flow i, the system would have more packets transmitted.

In order to maintain fairness among all flows, when there are backlogged flows with the best channel state, the scheduler should still consider the flows with channel states between the best and the worst to transmit even though such poor flows will realize a lower bandwidth utilization. This is the reason why keeping fairness and maximizing throughput are often conflicting goals. To tackle this challenging scheduling problem, we need to design a fairness notion such that the following two goals are achieved: 1.) flows with different channel states should not be treated as the same; and 2.) the throughput should not be degraded desperately.

17.3 SYSTEM MODEL

Figure 17.3 shows a single cell in a cellular phone network. The cell is managed by a base station (BS), which supports integrated video, voice, and file data services to mobile terminals (MTs) within the cell. Transmissions in the cell involve both the uplink (from a MT to the BS) and the downlink (from the BS to a MT). A scheduling algorithm is running at the BS, responsible for making scheduling decisions for both the uplink and the downlink traffic. In later discussion, we consider the scheduling of packets in a single cell only.

Multiple Access Scenario

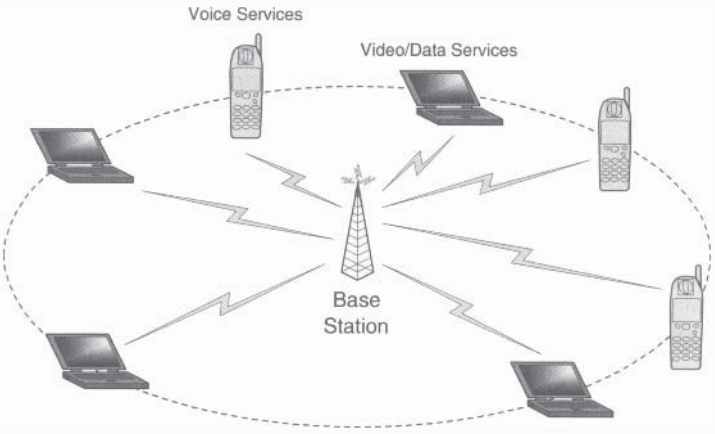


Figure 17.3 Packet scheduling in a cell.

Figure 17.4 shows the multiple access scenario in a wireless environment mentioned before. The system resource at the BS is limited. Different MTs contend for the shared channel, and signals from the admitted MTs are multiplexed and transmitted together by the BS. If $S(t)_i$ stands for the signals from MT i, then the overall received signal at the BS is:

$$y(t) = \sum_{i=1}^{N} a_i \cdot S(t)_i + n(t), \tag{17.1}$$

where a_i is the channel fading associated with the MT i, $n(t)$ is the thermal white Gaussian noise.

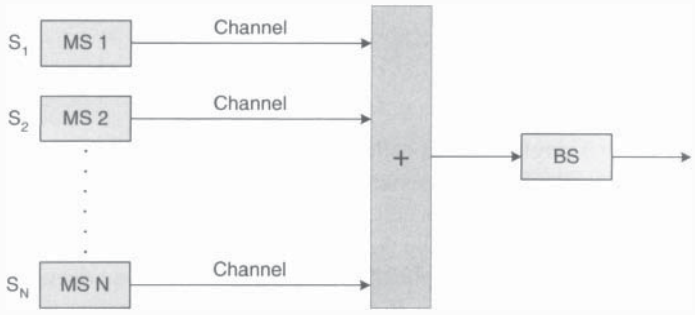


Figure 17.4 Multiple access scenario.

In order to ensure each intended receiver to extract its signal from the received $y(t)$ with multiplexed signals, given a frequency spectrum, the system resource must be divided into orthogonal channels. Each MT is assigned one or more orthogonal channels to transmit its packets, then the intended receiver can identify its signal because its channel is orthogonal to the others.

In general, there are three popular methods to separate the system resource into orthogonal channels: TDMA (Time Division Multiple Access), FDMA (Frequency Division Multiple Access), and CDMA (Code Division Multiple Access). A TDMA scheme divides the time line into slots. MTs can transmit data using the whole bandwidth in their scheduled slots. Different MTs occupy different time slots. A FDMA scheme partitions the whole frequency spectrum into channels. Each MT communicates on one certain channel with bandwidth throughout the communication, and the frequency channel is only reserved for this particular MT and no one else can share it. A CDMA scheme employs another dimension—spreading codes used in spread spectrum communications. Since the codes are orthogonal, channels encoded with different codes are separated from each other. Each MT can transmit using the whole bandwidth. In this chapter, we will mainly focus on discussing packet scheduling in TDMA based networks.

Figure 17.5 shows the TDMA frame structures for the uplink and the downlink in a cellular environment depicted in Figure 17.3. In the uplink, a frame is divided into three subframes. They are the request subframe, information subframe, and the pilot symbol subframe (note that pilot symbols are known reference symbols). Specifically, there are mini-slots in the request subframe for voice requests reservation and data requests contention. Note that a data request is not allowed to make reservation in the sense that even if a data request successfully seizes an opportunity to transmit in the current frame, it has to contend again in the next frame if it has some more data to send. There are information slots in the information subframe for the transmission of voice or data packets. Finally, there are slots in the pilot symbol subframe. A downlink frame is similarly partitioned into four subframes, namely the acknowledgment subframe, poll-for-CSI (Channel State Information) subframe, information subframe, and announcement subframe. The frame duration is 2.5 msec. Such a short frame duration has the advantage of having shorter delay and being practicable in wideband systems.

In TDMA based networks, the BS can transmit one packet at a time. On the contrary, in CDMA based networks, the BS can generally transmit more than one packet at a time because the system capacity is interference limited. As a TDMA based network can only support one MT at a time (at a particular time slot), wireless packet scheduling problem in TDMA based networks can be considered as a single-server problem. Figure 17.6 illustrates a packet scheduling scenario in a TDMA based network.

Due to the short propagation delay in nowadays wireless packet data networks (e.g., cellular networks or wireless LANs), we can normally assume that the MTs can almost immediately know the request result. This assumption can be justified

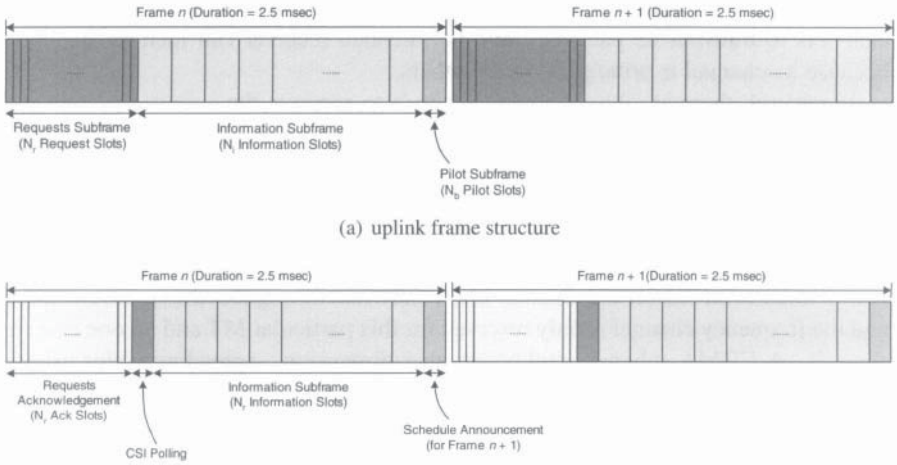


Figure 17.5 Uplink and downlink frame structure.

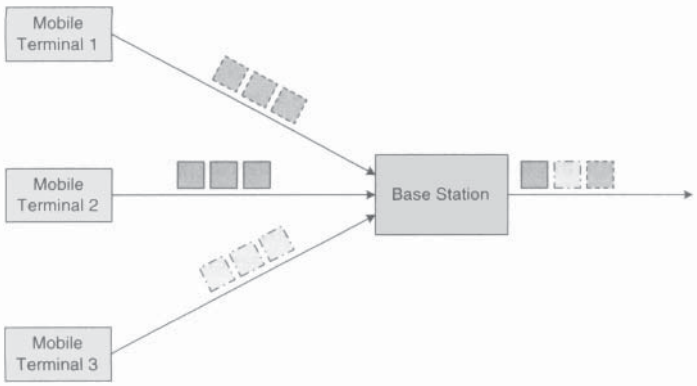


Figure 17.6 Packet scheduling in a TDMA based network.

by the following arguments. First of all, the delay involved from the time when the MTs send the requests to the BS to the time that the MTs realize the result of the contention (by checking the announcement on the downlink) is composed of two parts, namely the processing delay at the BS and the round trip propagation delay between the MTs and the BS. For high speed wireless access systems, the propagation delay is on the order of second (for a 30m separation), which corresponds to less than a mini-slot. The processing time required at the BS is also rather short because at the time of sending the acknowledgment to the MT, the BS does not need to make any decision on resource allocation yet. Thus, the request packet processing delay will be negligible (much shorter than a mini-slot). Considering the worst case, the delay of the acknowledgment is less than two mini-slots. Thus, after transmitting the request to the BS, the MT simply listens for subsequent downlink acknowledgment mini-slots (not necessarily the immediately following acknowledgment slot) with a time-out value (e.g., 5 mini-slots). The MT will retry sending another request when the timer expires.

Source Models:

Video and voice packets are assumed to be delay sensitive while data packets are assumed to be delay insensitive. Thus, voice packets are labeled with deadlines. A voice packet will be dropped by the MT if the deadline expires before being transmitted. Such packet dropping has to be controlled to be within a certain limit (e.g., below 1% as indicated in [137]) in order that the quality of service to the voice MTs is still acceptable. The source and contention models are summarized below.

Voice Source Model: The voice source is assumed to be continuously toggling between the talkspurt and silence states. The duration of a talkspurt and a silence period are assumed to be exponentially distributed with means t_t and t_s seconds, respectively (as indicated by the empirical study in [137], $t_t = 1$, and $t_s = 1.35$). It is assumed that a talkspurt and a silence period start only at a frame boundary.

Video Source Model: We use video teleconferencing as an example video source. Adapting the model used in [197], the number of packets per video frame period (i.e., 40 msec for a 25 fps frame rate) is governed by the DAR(1) model, which is a Markov chain characterized by three parameters: the mean, the variance, and ρ. The transition matrix is computed as: $P = \rho I + (1 - \rho)Q$, where ρ is the autocorrelation coefficient and I is the identity matrix. Furthermore, each row of Q is identical and consists of the negative binomial probabilities: $(f_0, ..., f_K, F_k)$, where $F_k = \sum_{k<K} f_k$, and K is the peak rate. Similar to a voice source, a video source can only tolerate a 1% packet loss rate [197].

Data Source Model: The arrival time of file data generated by an MT is assumed to be exponentially distributed with mean equal to one second. The data size, in terms of number of packets, is also assumed to be exponentially distributed

with mean equal to 100 packets. It is assumed that packets arrive at a frame boundary.

Request Contention Model: In a TDMA system, to avoid excessive collisions, even if a voice or data request has some packet waiting to be sent, the MT will attempt to send a request at a request mini-slot only with a certain permission probability.

In a TDMA system, an MT entering a new voice talkspurt or generating a new stream of data packets transmits an appropriate request packet in one of the request slots of the next frame. If there is more than one packet transmitted in the same request slot, collision occurs and none of the requests will be correctly received. At the end of each request slot, the successful or unsuccessful request will be identified and broadcast by the BS. An unsuccessful MT (does not receive the acknowledgment announcement in the downlink frame) can retry in the next request slot. On the other hand, a successful MT then transmits its information packets in the corresponding information slot in the current frame.

Channel Model:

Wireless channel capacities are time-varying and location-dependent. Modeling the wireless channel has historically been one of the most difficult parts of wireless system design.

Let $c(t)$ be the combined channel fading which is given by:

$$c(t) = c_l(t)c_s(t), \tag{17.2}$$

where $c_l(t)$ and $c_s(t)$ are the long-term and short-term fading components, respectively. Both $c_l(t)$ and $c_s(t)$ are random processes with a coherence time (time separation between two uncorrelated fading samples) on the order of a few milliseconds and seconds, respectively.

Short-Term Fading: Without loss of generality, we assume $E[c_s(t)^2] = 1$ where $E[x]$ denotes the expected value of a random variable x. The probability distribution of $c_s(t)$ follows the Rayleigh distribution given by:

$$f_{c_s} = c_s \exp(-\frac{c_s^2}{2}). \tag{17.3}$$

We assume that the mean and maximum speeds of the MT are 50 km/hr and 80 km/hr, respectively. Thus, the Doppler spread [32] is given by $f_d \approx 100Hz$. It follows that the coherence time, denoted by T_c, is approximately given by:

$$T_c = \frac{1}{f_d}, \tag{17.4}$$

which is about ten milliseconds.

Long-Term Fading: The long-term fading component, $c_l(t)$, is also referred to as the local mean [32] which, as shown by field test measurement, obeys the log-normal distribution:

$$f_{c_l}(c_l) = \frac{4.34}{\sqrt{2\pi}\sigma_l c_l} \exp(-\frac{(c_l(dB) - m_l)^2}{2\sigma_l^2}), \tag{17.5}$$

where m_l and σ_l are respectively the mean (in dB) and the variance of the log-normal distribution, i.e., $c_l(dB) = 20log(c_l)$. Since $c_l(t)$ is caused by terrain configuration and obstacles, the fluctuation is over a much longer time scale. Again, from field test results, the order of time span for $c_l(t)$ is about one second. Since MTs are scattered geographically across the cell and are moving independently of each other, we assume the channel fading experienced by each MT is independent of each other.

For error protection, redundancy is incorporated into the information packet. To exploit the time-varying nature of the wireless channel, a variable-throughput channel-adaptive physical layer [198] as illustrated in Figure 17.7 can be employed. CSI (Channel State Information), $c(t)$, which is estimated at the receiver, is fed back to the transmitter via a low-capacity feedback channel. Based on the CSI, the level of redundancy and the modulation constellation applied to the information packets are adjusted accordingly by choosing a suitable transmission mode. Thus, the instantaneous throughput is varied according to the instantaneous channel state. The essence of incorporating a channel-adaptive physical layer in the MAC protocol is to capture the notion of multiple channel quality levels such that the scheduler in the MAC protocol can exploit the scheduling diversity to optimize overall system performance. Besides using the above channel-adaptive technique, other channel-adaptive transmission models/techniques [356] can also be employed.

We assume the coherence time of the short-term fading is around ten milliseconds, which is much longer than an information slot duration (which is 2.5 milliseconds). Thus, the CSI remains approximately constant within at least two frames and it follows that the transmission mode for the whole frame is determined only by the current CSI level. Most importantly, scheduling decisions made at the beginning of a frame are based on accurate channel state information about the frame yet to be transmitted. Specifically, transmission mode q is chosen if the feedback CSI, $\hat{c}$, falls within the adaptation thresholds, (ξ_{q-1}, ξ_q). In the case of ABICM, the operation and the performance of the scheme is determined by the set of adaptation thresholds, $\{\xi_0, \xi_1, ...\}$. If the ABICM scheme is operated in the constant BER (bit-error rate) mode, then the adaptation thresholds are set optimally to maintain a target transmission error level over a range of CSI values. When the channel state is good, a higher mode could be used and the system enjoys a higher throughput. On the other hand, when the channel state is poor, a lower mode is used to maintain the target error level at the expense of a lower transmission throughput. Note that when the channel state is very bad (e.g., a very low SINR is perceived), the adaptation range of the ABICM scheme can be

exceeded such that the throughput (mode-0) becomes so low, making it impossible to maintain the targeted BER level, and thus, the mobile terminal (or the flow) cannot transmit any packet.

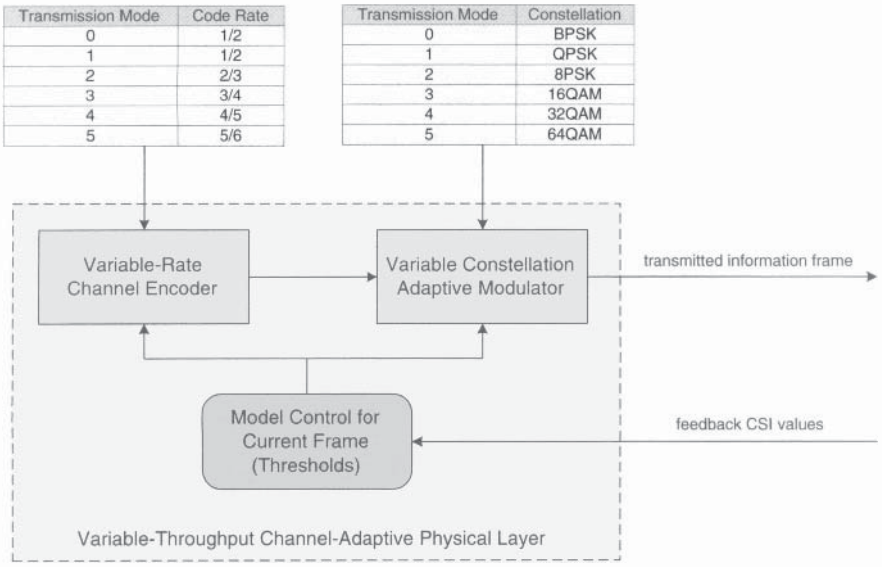


Figure 17.7 A conceptual block diagram of the variable-throughput channel-adaptive physical layer [198].

Given the above considerations about the channel state, the instantaneous throughput offered to the access control layer, denoted by ρ, is also variable and is therefore conceptually a function of the CSI, $c(t)$, and the target BER, P_b, denoted by $\rho = f_\rho(c(t), P_b)$.

17.4 FAIRNESS NOTIONS

In a wired network, all the backlogged flows constitute the scheduling set. The scheduler tries to allocate the bandwidth among all the backlogged flows proportionally to their preallocated service shares. Whenever the system allocates some bandwidth to a certain flow, it will achieve a certain throughput accordingly. This is because in the wired line transmission, the efficiency (the bandwidth utilization) of using the channel is constant among all flows.

However, in the wireless environment, the backlogged flows have different channel states. There are flows which can make full use of the bandwidth, flows which cannot transmit any packet, and flows with channel states in between. The scheduling set is

different from that in the wired network's case. For example, the flows which cannot transmit any packet should not be scheduled even they are backlogged.

In a wireless environment, the scheduling set is a subset of a typical scheduling set in a wired network. Moreover, this subset is also variable over time because the channel states vary as time goes by. However, the scheduling set in the wired networks includes all the backlogged flows, and thus, if the wired fairness notions are employed, a scheduler will try to allocate the same throughput level to all the backlogged flows. However, in wireless networks, for the flows which can transmit no packet, the system will simply keep on wasting the resource in vain and possibly starve the other flows. This is inefficient from a resource utilization point of view because those flows suffering from deep fading (i.e., channel quality is not good—with a very low SINR) will not be able to utilize the time slots efficiently (e.g., data loss may occur more frequently). A more intelligent method is to allow the flows having better channel states to proceed first. This is the motivation for designing fairness notions that are adaptive to the wireless environment.

Greedy and Realistic Fairness Notions:

There are several existing fairness notions for the wireless environment, and they can be classified into two categories:

1. For fairness notions which are designed with the assumption of the two-state Markov chain channel model, they are regarded as greedy fairness notions because the scheduler will only schedule the flows with "good" channel state to transmit so that it can make use of the channel greedily;

2. For fairness notions which are designed with the assumption of a realistic channel model, such as the one mentioned in Section 17.3, they are regarded as realistic fairness notions because they are more useful in a realistic wireless network.

As for the greedy fairness notions, the scheduling set includes only the backlogged flows with the best channel state. The common drawbacks of the greedy fairness notions are:

- they are unfair to the flows with channel states between the worst and the best; and

- they are not realistic—flows can spend most of the time in channel states between the best and the worst, and they receive no service, while flows with the best channel state will not relinquish their service to help these unlucky flows catch up.

In the realistic fairness notions, the scheduling set includes all the backlogged flows except those which cannot transmit any packet. Thus, the scheduling set is larger, and more realistic and fair. But the channel states of the flows in the scheduling set are not the same. This makes the design of fairness notions more complicated. For

example, we need to decide if flows with different channel states should receive the same normalized amount of time slots or realized throughput. Note that the system will need to allocate much more time slots to the flow with a bad channel state to achieve the same amount of throughput with the flows with a good channel state.

Effort and Outcome Fair:

In a broad sense, fairness can be defined with respect to two aspects: effort and outcome [375]. Intuitively, a policy is called effort fair if the allocation of services to different flows is fair, without regard to the actual amount of data successfully delivered by the flows using the allocated services. Informally, "fair" means a flow gets the service amount that it deserves to get. On the other hand, a policy is called outcome fair if the actual realized data throughput among the flows is fair.

Effort Fair: A scheduler is fair if the system resource (e.g., the bandwidth or time slots in TDMA based network) the system allocates to different flows is proportional to the different service shares of the MTs. Mathematically, that means the difference between the normalized services the system allocates to any two backlogged flows i and j is bounded as follows:

$$\left| \frac{S_i(t_1, t_2)}{r_i} - \frac{S_j(t_1, t_2)}{r_j} \right| < \varepsilon, \tag{17.6}$$

where $S_i(t_1, t_2)$ denotes the allocated service of a certain flow i during time interval $(t_1, t_2]$, r_i is the requested service share, and ε is a finite constant. Such a fair scheduler can be considered as effort fair [293] in the sense that the scheduler only guarantees the effort expended on the flows is fair, without regard to the actual throughput achieved by the different flows.

Outcome Fair: A scheduler is fair if the difference between the normalized amount of realized throughput of any two flows i and j is bounded as follows:

$$\left| \frac{T_i(t_1, t_2)}{r_i} - \frac{T_j(t_1, t_2)}{r_j} \right| < \varepsilon, \tag{17.7}$$

where $T_i(t_1, t_2)$ denotes the actual throughput a flow i achieves during the time interval $(t_1, t_2]$. Such a fair scheduler can be considered as outcome fair [52] in that the scheduler tries to provide a fair actual performance achieved by the flows (rather that the "nominal" performance as in the effort fair definition discussed above).

In a TDMA based network, "effort" refers to the allocated bandwidth (time slots) and "outcome" refers to the realized throughput using the allocated time slots. Note that a variable effective throughput is manifested by the fact that some data may be lost due to poor channel states and thus, inducing retransmissions; or in adaptive FEC schemes such as ABICM, the amount of data protection varies according to the channel states.

Outcome fair and effort fair are equivalent in wired networks. However, in wireless networks, more error protection will be added and the realized throughput will be reduced when a flow does not have a perfect channel state, which results in a discrepancy between "effort fair" and "outcome fair". In other words, there is much difference between allocating the same amount of time slots among MTs and realizing the same amount of throughput among MTs. The question remained is what kind of fairness should be maintained among the MTs with the consideration of their channel states.

The efficiency of using the bandwidth should be considered when trying to allocate time slots "fairly". MTs with different channel states should not be treated as the same (i.e., neither effort fair nor outcome fair should be maintained among MTs with different channel states) because they have different efficiency of using the bandwidth allocated. That is the reason why it is challenging to have a reasonable fairness notion which can be applied in the scheduling set with different channel states, while it greatly simplifies the scheduling problem when the scheduling set includes only the flows with "good" channel state as in greedy fairness notions.

Now, there is a dilemma: 1.) we can make use of the bandwidth greedily to improve efficiency of the system by giving the flow with the best channel state the highest priority but this is unfair to those flows which spend most of the time having poorer channel states; or 2.) we can help to maintain outcome fair at the expense of wasting the bandwidth but this is unfair to those flows with better channel states because they are allocated less time slots.

Another aspect worth noting is that a reasonable fairness notion should not lead to severe degradation of the overall throughput of the system. That is the reason why outcome fair is inapplicable. Indeed, if outcome fair is strictly enforced, some MTs with especially bad channel state will occupy a lot of time slots and have a very low effective throughput. This conflicts with the task of the scheduling algorithm of making use of the system resource intelligently (a system side performance metric) while keeping the fairness property (a user side performance metric).

Comparison of Fairness Notions:

Figure 17.8 shows the classification (according to the fairness notions employed) of a number of existing wireless packet scheduling algorithms: WPS (Wireless Packet Scheduling) [227], IWFQ (Idealized Wireless Fair Queueing Algorithm) [227], CIF-Q (Channel-Condition Independent Fair Queueing) [257], SBFA (Server Based Fairness Algorithm) [293], WFS (Wireless Fair Service) [254], CS-WFQ (Channel State Independent Wireless Fair Queueing) [215], ELF (Effort Limited Fairness) [100], PF (Proportional Fairness) [168], and CAFQ (Channel Adaptive Fair Queueing) [344]. The details of the algorithms are described in Section 17.5.

Table 17.1 gives a qualitative comparison of different fairness notions employed in scheduling algorithms mentioned previously.

In CIF, the need to maximize the total throughput never gives in. Thus, "swap" or "borrowing" takes place to use the channel effectively at the expense of losing the instant fairness among the MTs. But in the long term, outcome fair is achieved. Bias

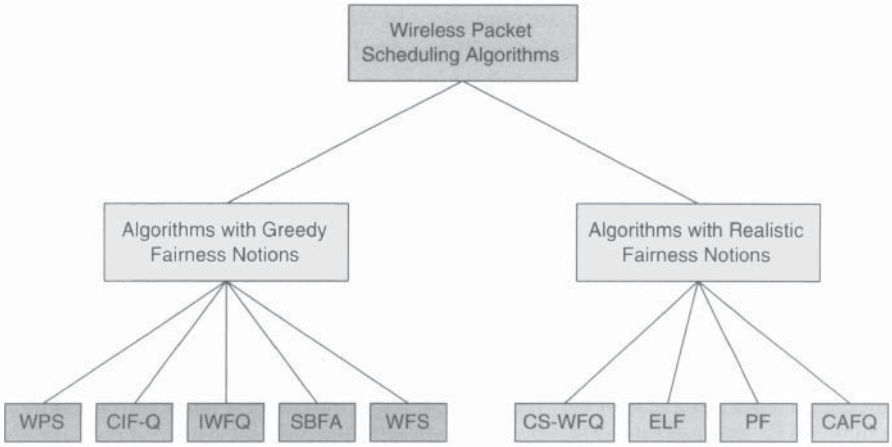


Figure 17.8 Classification of existing wireless packet scheduling algorithms.

Table 17.1 Comparison of different fairness notions.

Fairness	Short-Term	Long-Term
CIF	Short-term fairness is maintained only among the backlogged flows with perfect channel states; neither outcome fair nor effort fair is considered for flows with "not so good" channel states.	Outcome fair
ELF	Outcome fair is maintained among the backlogged flows with channel states better than their predefined thresholds.	Not precisely defined
PF	Short-term fairness is not precisely maintained.	Not precisely defined
CAF	Short-term fairness (normalized by channel states) is maintained among all the backlogged flows except the ones with the worst channel state. A balanced consideration of fairness and system throughput is achieved.	Outcome fair

against the flows with non-perfect channel states exists, and the fairness notion is in favor of the flows with good channel state and maximizing the throughput.

In ELF, the compromise is made between maximizing the throughput and keeping fairness by setting thresholds. When the channel states are within the threshold, output fair is kept at the expense of wasting the bandwidth. There is no any endeavor made to save the throughput of the system unless the channel state is poorer than the threshold. And the worst feature is: there is no action taken to maintain fairness in the long run.

In PF, the adjustment between the two goals is made by putting the channel state element on the numerator and the realized throughput element on the denominator

in the priority calculation. However, the fairness notion is not strict and there exists pathological situations.

In CAF, the balance between the two needs can be tuned by a punish factor at any time. The punish factor can help to decide between when to make use of the bandwidth more efficiently and when to treat every flow more "fairly". When the punish factor is set to be very large, CAF will be more like the greedy fairness notions. Flows with bad channel states have little chance to access the bandwidth. Then the system can maximize the throughput. When the punish factor is -1, CAF will behave in the same manner as ELF when the channel state is within the threshold. The system tries to keep output fairness among MTs at the expense of degrading the throughput. For a punish factor larger than -1, the punish factor enables the system to schedule the flows with bad channel states to keep fairness at the same time of punishing them and saving the throughput.

17.5 FAIR QUEUEING APPROACHES

In this section, we discuss several existing packet scheduling algorithms: CIF-Q [257], ELF [100], PF [168], and CAFQ [344].

CIF-Q (Channel-Condition Independent Fair Queueing):

In [257], Ng *et al.* define the notion of Channel-Condition Independent Fair (CIF) such that the scheduler can:

1. provide delay and throughput guarantees for the error-free flows;
2. provide long term fairness: a lagging and continuously backlogged flow which enjoys an error-free channel after some time will be guaranteed to catch up within a certain time bound;
3. provide short-term fairness: the difference between the normalized services received by two error-free flows which are in the same state (lagging, leading, or satisfied/in-sync) during a time interval is bounded; and
4. provide graceful degradation for the leading flows: a leading error-free flow will only give up part of the service it would receive in the companion simulated error-free system to compensate the other lagging flows (i.e., compensation is done in a gradual manner; in some other algorithms [227, 254], once a lagging flow returns to a good channel state, it may get back all its missing services and hence effectively shut down the leading flows).

CIF can be useful in a wireless environment as it takes the non-perfect channel state (i.e., burst errors occur) into consideration. It allows the scheduler to delay the service of a flow if it does not have a good channel state, and the scheduler will try to help this flow to catch up in the long run. However, because the CIF-Q scheduler does not allow a flow having "intermediate" (not so good and not so bad)

channel states to transmit, the scheduling set contains only flows with "good" channel state. As we have emphasized earlier, this scheduling philosophy is unsuitable in a practical environment, where we can expect that a significant portion of flows will have intermediate channel states because experiencing moderate levels of burst errors (that can be corrected by appropriate FEC's) is the norm rather than the exception in a wireless network.

The general drawback in CIF-Q is that in the short term, where the system gives overwhelming priority to the flows with good channel state, while all the other error flows (with possibly different channel states that are only slightly worse than the perfect channel state) are just treated as the same: not allowed to transmit. This is unfair among flows and, most importantly, may unduely increase the average delay of many flows. This is unacceptable for supporting isochronous media data.

In fact, a lagging flow can only be guaranteed to catch up when it has an error-free channel afterwards. However, in reality, the time period during which a flow can make full use of the bandwidth is limited. In other words, a flow can spend most of the time in channel states between the perfect and the worst. An efficient scheduling algorithm should provide fairness and performance guarantee even while the channel states of the flows are varying among different quality levels. These drawbacks in CIF-Q can also be found in other scheduling algorithms with greedy fairness notions.

ELF (Effort Limited Fair):

In ELF scheduling [100], it sets the weight of each flow dynamically. By adjusting the weight of each flow according to its channel state, ELF schedules the utilization of time slots among all the backlogged flows except the ones with the worst channel state. The poorer the channel state that a flow perceives, the higher the weight is, and the more time slots the flow is allocated. Thus, ELF scheduling tries to maintain outcome fair among flows with channel states better than their pre-set thresholds. When a flow has a channel state even poorer than its threshold, the system will not go on increasing its weight. Instead, it will be allocated the same weight as when it has a channel state right at the threshold, and thus, the system will not increase the portion of time slots the poor flow occupies. Therefore, there is an effort-limit for each flow, and outcome fair is maintained for flows with channel states better than their threshold. Within the limit, the poorer the channel state a flow has, the more effort is exerted on this flow. But the effort is limited, so no more effort will be paid beyond this limit.

A power factor P_i is used as an administrative control parameter to decide the channel state threshold, and the weight bound. It is used to adjust the weight as follows:

$$\bar{W}_i = \min(\frac{W_i}{1 - E_i}, P_i \cdot W_i), \tag{17.8}$$

where $\bar{W}_i$ stands for the adjusted weight of flow i, W_i is the original pre-allocated weight of flow i, E_i is the error rate of flow i (e.g., a flow with 30% error rate can make use of 70% of the available bandwidth). For example, if the power factor is set to be 3, then when the error rate is 50%, the new adjusted weight is $2 \cdot W_i$; when

the error rate is 66.7%, the new adjusted weight is $3 \cdot W_i$. Thus, as the flow changes from a better channel state to a poorer one, the adjusted weighted will increase so that this flow will be allocated more time slots to achieve outcome fair. However, when the error rate changes from 66.7% to 75%, the weight is bounded by $3 \cdot W_i$ and will not increase. The power factor helps to decide the channel state threshold below which the system will not increase the flow's weight more. The error rate threshold is: $E_i = 1 - \frac{1}{P_i}$.

The power factor is used to separate the outcome region and effort region, or to decide the "effort limit". With a realistic channel model, flows with different channel states will have different efficiency of using the bandwidth. All the flows with channel states better than the worst case have chance to transmit packets. This is more reasonable than the algorithms with greedy fairness notions. Among flows with channel states better than their threshold, outcome fair is maintained at the expense of a low efficiency of using the bandwidth. Effort limit is brought in order to prevent some flows with extremely or continuously bad channel states from monopolizing the bandwidth and demoting the whole performance. Effort limit fair is therefore useful in the realistic channel model and is easy to implement.

However, although there are channel state thresholds to limit the effort, for flows with channel states better than the thresholds, the poorer the flow's channel state is, the more time slots the system gives to the flow. This is unfair to the flows with better channel states. Other problems of the ELF approach include: there is no flow which can take leading and return service to lagging ones later, and there is no pre-allocated bandwidth to help the lagging flows to catch up.

PF (Proportional Fair):

Recently, designed for HDR (high data rate) services in CDMA systems, proportional fair [31, 168] is considered to be a simple yet effective fairness notion. Proportional fairness is formally defined in [187] that a scheduling decision (the bandwidth allocation) is proportional fair if using any other possible allocation will result in a non-positive normalized amount of aggregate allocation. At the scheduling time (time = t), suppose a flow i has an average effective throughput $H_i(\tau)$ over a past time window of length τ (i.e., from time $t - \tau$ to τ; average is calculated using low pass filtering), and the real throughput that can be achieved by flow i at time t is $\lambda_i(t)$, which is the aggregate rate of a certain number of supplemental channels (SCH) determined by the BS according to the interference and power limits. A scheduler can achieve proportional fair if it selects for transmission the flow with the largest value of $\frac{\lambda_i(t)}{H_i(\tau)}$ [168].

Thus, intuitively, a proportional fair scheduler heuristically tries to balance the services (in terms of outcome) of the flows, while implicitly maximizing the system throughput in a greedy manner. Obviously, the proportional fairness notion is a purely outcome fairness metric. Thus, while such a metric is simple to use, proportional fairness does not guarantee fairness in a strict sense. For example, consider the situation where a flow has experienced a prolonged period of poor channel states

(hence, has a small value of $H_i(\tau)$), it may not get service even though its channel state improves (e.g., with a moderately large value of $\lambda_i(t)$) if there is a "dominant" flow which has a very good channel state (i.e., a very large value $\lambda_i(t)$).

Another problem is that the lagging flows are not given more consideration. In fact, flows are not classified by leading or lagging in proportional fair queueing. Flows, which have fulfilled their QoS requirements, will not receive less service even when there are other flows which have not fulfilled their QoS requirements. For example, before time t, flow i has received more service than its QoS requirement (we name it as the leading flow) because it has a good channel, however, flow j received much less service than its requirement (we name it as the lagging flow) because it has an especially bad channel before. Between time t and $t + \tau$, flow i has a bad channel and receives less service. At time $t + \tau$, even if the channel state of flow j is better than that of flow i, flow i may still be chosen to occupy the channel as it has a low $H_i(\tau)$ (at this time, it is possible that flow i is seriously leading and flow j is seriously lagging). This is unfair. Furthermore, the delay experienced by flows can also be high in a proportional fair system.

CAFQ (Channel Adaptive Fair Queueing):

CAFQ employs the fairness notion CAF (channel adaptive fairness) [344], and has the following features:

1. contrary to CIF-Q, graceful degradation is not ensured, in order to help the lagging flows more efficiently;
2. a punish factor is used to decide how seriously the scheduler punishes a non-perfect channel state flow that transmits packets; and
3. a virtual compensation flow is incorporated to help the lagging flows to catch up.

From the MT's viewpoint, fairness should be maintained in that so long as a flow can transmit some data, it should be provided some chance to transmit. At the same time, QoS should also be met. However, the BS can transmit one packet at a time in TDMA based networks, so from the system manager's viewpoint, it must improve the efficiency of using the channels to meet the QoS. The goal to keep fairness often conflicts with the goal to improve the efficiency of using the channels in face of the time-varying quality of the channels. Because whenever a flow without a perfect channel state is allowed to transmit, there will be part of the bandwidth wasted, and the wasted bandwidth can never be replenished. It should be noted that this is very different from the idea of swapping flows that are error-free and error-prone, as in existing scheduling algorithms such as CIF-Q. When an error-free flow takes the opportunity of an error-prone flow, it will relinquish the service when the error prone one recovers.

If abundant bandwidth is available or the channel state is most likely to be perfect, we should maintain the graceful degradation, and prevent the non-lagging ones from starving. But in a realistic system in which the channel is not so good, we cannot

expect to achieve perfect allocations, but rather we should meet the flows' QoS first. Thus, in CAFQ, graceful degradation is not implemented and the rationale is to compensate the lagging flows as soon as possible. This can help to achieve the QoS requirements of isochronous media data types.

In CAFQ, the virtual compensation flow is incorporated with the following properties:

- swapping of flows is used at the same time of the virtual compensation flow to compensate the lagging flows;
- CAF is strictly maintained in the short term, and the virtual compensation flow helps to keep outcome fair in the long term; and
- the flow which is most seriously lagged has the highest priority of receiving service from the virtual compensation flow.

17.6 PRACTICAL ILLUSTRATIONS—HSDPA

To exploit the advantages of channel adaptation and multiuser selection diversity, the high-speed downlink packet access (HSDPA) scheme is incorporated in a more recent version of the UMTS standard—Rel 5. The HSDPA system adopts a microscopic scheduling approach on the radio resource (timeslots/codes) for its serving active users over very short duty cycles (up to 2ms). To facilitate the microscopic scheduling, the scheduling algorithm resides at the base station instead of the RNC to minimize the potential delay in the execution of scheduling. The base station obtains the instantaneous channel quality estimates from the mobiles (UEs) and selects one mobile (UE) to transmit on the high-data-rate shared-traffic channel at the current time slot, based on a scheduling algorithm (which factors in system throughput, QoS requirements, fairness, or a combination of these). This is fundamentally different from the Rel 99 approach. For example, if there are 10 packet-switched data users in the cell, there will be one high-data-rate traffic channel shared dynamically between these 10 users in the HSDPA systems. In fact, it is shown that the microscopic scheduling approach in the HDSPA systems is theoretically optimal because of the multiuser selection diversity.

17.7 SUMMARY

In this chapter, we have discussed the packet scheduling problem in wireless networks, and we have also talked about the reasons why the extensively explored fair queueing algorithms designed for the wired networks cannot be adopted to the wireless environment. Specifically, we discussed in detail on the properties of the wireless channel. Afterwards, different fairness notions and different wireless packet scheduling algorithms were discussed.

PROBLEMS

17.1 Briefly discuss the difference of packet scheduling in wired and wireless networks.

17.2 As for packet scheduling, what is the problem of using a two-state Markov chain channel model?

17.3 What is the difference between greedy fairness notions and realistic fairness notions? How will the difference affect the packet scheduling problem?

CHAPTER 18

POWER MANAGEMENT

Power is the first good.

—R. W. Emerson: Inspiration, 1876

18.1 INTRODUCTION

Power management is an important issue a wireless system as the devices are usually battery-powered. Indeed, power conservation is sometimes the top priority task in extending the lifetime of the mobile terminals and also that of the whole network.

Interoperability:

We first discuss the power consumption characteristics of the mobile terminals in Section 18.2. With the understanding of power consumption characteristics, it can be concluded that power can actually be conserved in two different situations. The first situation is that the mobile terminal is under active communication and the second situation is that the mobile terminal is under idle time. In the first case, power is generally conserved by adjusting the transmit power levels and selecting

Wireless Internet and Mobile Computing. By Yu-Kwong Ricky Kwok and Vincent K. N. Lau

Copyright © 2007 John Wiley & Sons, Inc.

an appropriate terminals for relaying the packets. These techniques will be briefly discussed in Section 18.3.1. On the other hand, when the terminals are not idle, they can usually be changed into low-power state to further conserve energy. However, they have to wake up periodically to check for packets destined for them. These techniques will be further discussed in Section 18.3.2. Most of the techniques suggested in active communication and that of idle time can actually be used simultaneously to conserve power.

Performance:

Various power conservation schemes are suggested to conserve power during active communication and during idle time. However, as we will see in this chapter, most of the schemes in fact will sacrifice deteriorate the throughput and delay of the network as a tradeoff for power conservation. Even worse, some of the power conservation schemes may actually increase the power consumption. These will be further discussed in Section 18.4.

18.2 CHARACTERIZATION OF POWER CONSUMPTION

In many wireless ad hoc networks, mobile terminals are largely battery powered. Thus, power conservation become an important issue to extending the lifetime of the mobile terminals, and, in return, prolong the lifetime of the network. Before various power conservation schemes are discussed in detail, the power consumption characteristic of mobile terminals will be briefly discussed. Most of the power conservation techniques try to save power based on these characteristics.

As shown in Figure 18.1, mobile terminals spend their power differently in various states and various layers. At the beginning, the mobile terminals in ad hoc networks can be in one of the four states—transmit, receive, idle and sleep. In these states, the energy dissipation is very different. The energy dissipation of mobile terminals in transmit mode is around 1.5 to 3 times that of the receive mode. While the energy dissipation of mobile terminals in transmit mode is around 5 to more than 30 times that of the sleep mode. The energy dissipation of idle mode is somewhere in-between receive mode and sleep mode. However, most specifications do not distinguish energy dissipation between receive mode and idle mode [247].

Moreover, energy is also required for a mobile terminal to actually put information in the wireless medium in the transmit mode and get information from the wireless medium in the receive mode. These constitute the power consumption in physical layer. Generally speaking, it is possible for the mobile terminal to select transmitting power level based on the actual environmental characteristics such as the transmission rate, the received SNR, the distance between the transmitter and receiver, etc. Generally, received power is proportional to the transmitted power and inversely proportional to the square of the distance between transmitter and receiver[1] The higher the

[1]Propagation model in free space.

received SNR, the more complex modulation coding can be used to encode more bits into one symbol. In this case, less transmission is needed to transmit the information to the other side [247].

In MAC layer, energy is also required for the sender and receiver to compete for the shared wireless medium channel with other mobile terminals in the network. In DCF mode of IEEE 802.11, four time frame sequence is generally used. In this sequence, the sender terminal needs to send out RTS and the DATA, while the receiver terminal needs to send out CTS and the ACK. On the other hand, power is also required to retransmit the corrupted frames due to propagation error and packets collision with other terminals.

In wireless ad hoc networks, mobile terminals not only need to send out their own packets, but also constantly need to forward packets for other mobile terminals. It is because in such an environment, it is generally impossible for a mobile terminal to directly send a packet to its destination. In this case, it is required for the source mobile terminal to rely on some intermediate terminals to route its packets to destination. The routing of packets use up the power in the intermediate terminals and the selection of different intermediate terminals set will result in different power usages in the network view.

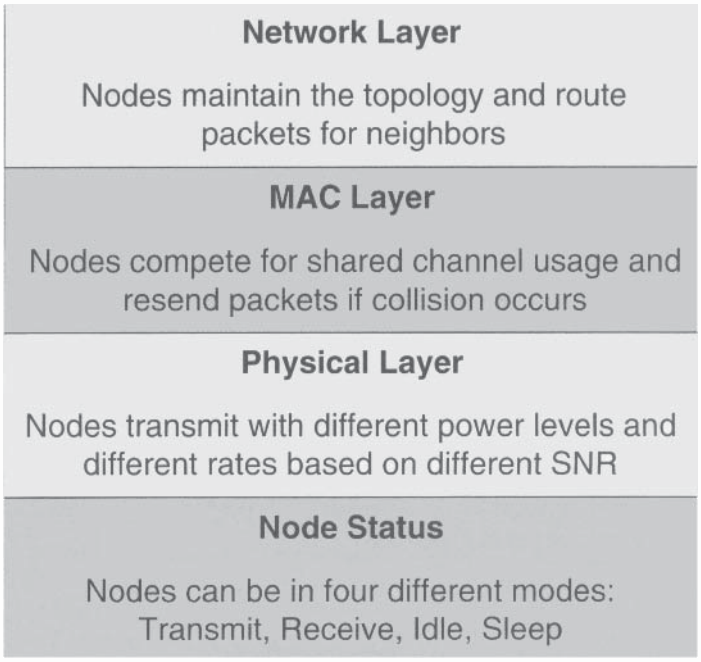


Figure 18.1 Power consumption characteristics.

It is thus obvious that power is consumed in different situations and under different extents. As a result, various power conservation schemes are developed in exploiting power consumption in scheduling of terminals in various states.

18.3 POWER CONSERVATION SCHEMES

We have seen that mobile terminals can be set into multiple states in which different power consumption are resulted. Specifically, during active communication—transmit and receive data, power will be consumed in various layers differently according to the approaches used in these layers. These lead to the development of two main categories of power conservation schemes—conservation under active communication and conservation under idle period [247].

Briefly speaking, power conservation schemes under active communication are based on two main techniques in conserving power for a particular device and/or devices in the whole network. They are power control which mainly target for Physical and MAC layers, and energy-conservation routing which mainly targets for network layer. This will be discussed in Section 18.3.1.

On the other hand, mobile terminals may not be in use all the time. In many practical situations, most of the time, they are idle in the network. Notice that power consumption in sleep mode is much lower than that in transmit, receive or idle mode. Consequently, it is reasonable to conserve power by putting those devices which are not under active communication into sleep mode. However, extra energy and time will be required to bring those sleeping devices into active again. Thus, power can only be conserved by putting those devices which will not be used soon into sleep mode. On the other hand, in order to have a functional network, only parts of the devices can be put into sleep during certain time slots so that the network is not partitioned. These issues are further elaborated in Section 18.3.2.

18.3.1 Power Conservation Schemes under Active Communication

In this section, two main types of power conservation schemes under active communication will be discussed in detail. They are power control techniques and energy-conservation routing techniques. Power control techniques are mainly focused in power conservation for a particular mobile terminal or a particular pair of mobile terminals by adjusting their transmitting power levels. On the other hand, energy-conservation routing algorithms are mainly focused on power conservation for the route paths from sources to destinations. As a result, the network lifetime can also be prolonged with the consideration of the paths of routing in the network [247].

18.3.1.1 Power Control Power control refers to the adjustment of the transmit power level by a mobile terminal which is designed to support transmit power level adjustment to transmit data to the neighbor node. As discussed in the previous section, received signal power level is directly proportional to the transmit power level

and inversely proportional to the square of the distance between the communicating nodes (assume in free space propagation model). In this case, for a given threshold received SNR for correctly decoding the messages, the shorter the distance between the communicating nodes, the smaller the transmit power level which is required to achieve this threshold.

There are two main advantages in using this approach. The first is that power can be conserved by using just about right transmit power level to send data to the receiver instead of using the maximum/normal transmit power to send data all the time. The second is that from the network's perspective, spatial reuse can be achieved much more effectively as illustrated in Figure 18.2. Here, nodes A, C, E are planning to transmit to nodes B, D, F, respectively. If normal transmit power is used, only one pair of nodes can transmit data at a time due to the four time frame sequences (i.e., RTS, CTS, DATA, ACK). However, if controlled transmit power is used, these three pairs of nodes can transmit simultaneously. As a result, spatial reuse can be achieved with the use of controlled power.

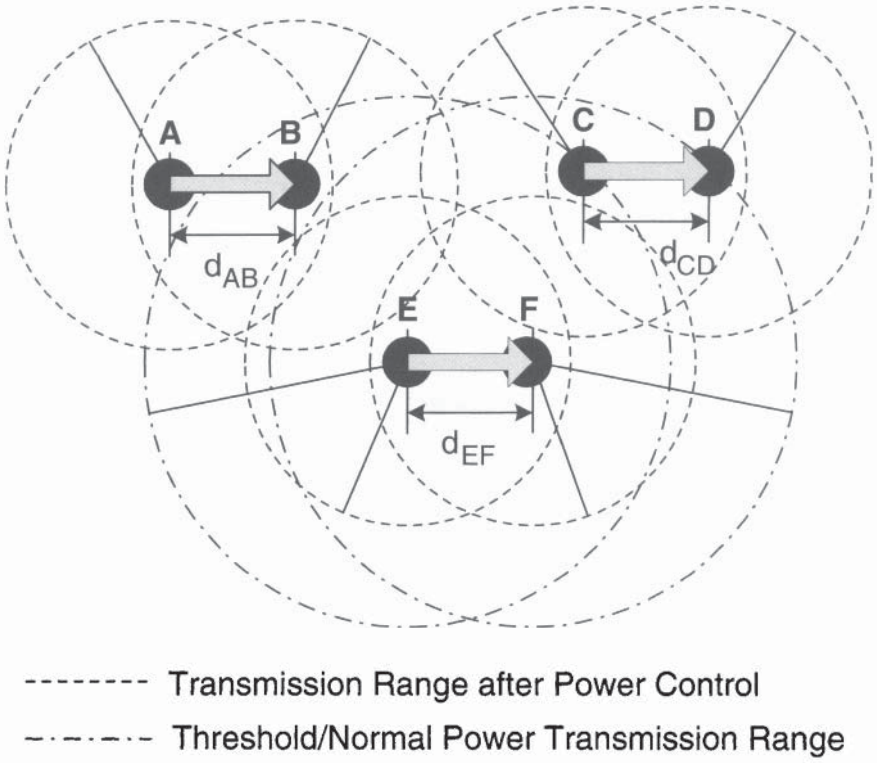


Figure 18.2 Power control example.

The main idea behind power control scheme is to use just about right power level to communicate with the neighbor nodes instead of using a standard, fixed, maximum power level to communicate with all the neighbors all the time. As a result, the power needed to communicate with near neighbor nodes can be greatly reduced and this results in power conservation for the network as a whole. Nevertheless, most of the power control schemes also lead to another problem—asymmetrical link problem as shown in Figure 18.3 [247, 177, 216].

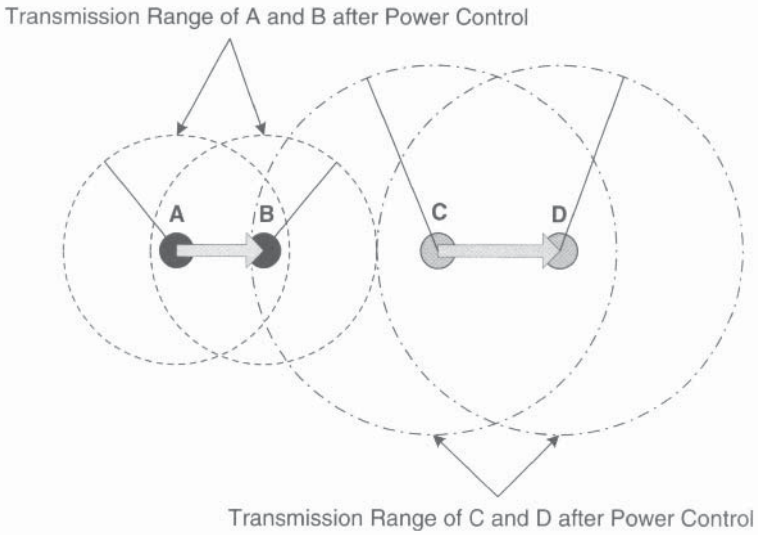


Figure 18.3 Asymmetrical link problem.

As can be seen, node A is communicating with node B, and node C is planning to start a communication with node D. Node A sends the data to node B using the about right power level and this results in a transmission range which cannot be detected by node C. At this moment, node C would like to initiate a communication with node D, so it starts by sensing the medium (refers to CSMA/CA in Section 9.4. However, as the transmission range of node A is much reduced by power control scheme and thus, its transmission cannot reach node C. Node C detects a free medium and starts transmitting data to node D using another power level which is high enough to reach node B as shown in Figure 18.3. As a result, a collision is likely to occur at node B if the power level of node C is high enough to corrupt the data of node A at node B. This greatly reduces the effects of power controls and may even *increase* power consumption, as more power will be consumed in retransmission.

In view of this problem, researchers have proposed to transmit RTS and CTS at normal power level, and transmit only DATA and ACK at reduced power level (refer to four-time-frame-sequences in Section 9.4). By doing so, packet collision is greatly

reduced. This is because the medium is first reserved by RTS-CTS frames. All the nodes surrounding the communication pair which can successfully decoded[2] the RTS-CTS frames will be able to set the NAV flags to the proper duration as shown in Figure 18.4. As can be seen, nodes C, D and G are within the decoding range of node A and B normal power level. In this case, the RTS and/or CTS can be decoded and detected by these nodes and so they can set the NAV accordingly. For nodes E and F, they can only sense the RTS and/or CTS of node A and B, but cannot decode them. Consequently, after the RTS and CTS transmissions, nodes A and B communicates (DATA and ACK) at a reduced power level. At this moment, if node E or node F would like to transmit packets, it is still possible to cause collision in node A or node B.

Motivated by this observation, a novel scheme called PCMA [248, 216, 247], which is based on the idea from dual bust tone multiple access (DBTMA) [85], is suggested. PCMA relies on the use of a second channel to transmit busy tone to the surrounding senders. Specifically, before the transmission of the frames, the sender can use the busy tone information to determine the transmit power level which will not cause collisions to the existing transmission. Thus, PCMA can effectively reduce data collision at the receiver side. Apart from the use of second channel into the protocol, PCM [177, 247] is based on the use of maximum transmit power level to transmit RTS and CTS. Then, alternately, longer reduced power and shorter maximum power are used to transmit DATA, and reduced power will be used to transmit ACK. This approach can also prevent the senders from missing out the RTS and CTS messages and subsequently consider the medium as free accidentally. However, since the ACK is also transmitted at reduced power in both approaches, ACK collision at sender side is still possible. In fact, it is very difficult to design a perfect MAC protocol which can improve both the capacity and power utilization at the same time.

18.3.1.2 Energy-Conservation Routing In an ad hoc network, it is usually impossible for the source node to directly send a packet to the destination node. As a result, some intermediate nodes have to be selected to help in routing the packets to the destination. Energy-conservation routing can generally conserve power of nodes in the whole network by selecting the minimum power consuming path (minimum energy cost) or selecting the maximum residual power path (maximum battery capacity) or a mix of both (minimum energy cost and maximum battery capacity). There are generally some advantages and disadvantages of these approaches, however. For example, the use of minimum energy path all the time may result in a swift depletion of energy of those mobile terminals on that path, leading to partitioning of the network. By contrast, the use of maximum battery capacity path may also result draining out the battery power of some nodes prematurely since the transmission costs are differ-

[2]For the same power level, there are two ranges. The nearer range is the decoding range and the farther range is the carrier sensing range. Nodes within decoding range can decode the messages sent by the originating node, while the nodes within the carrier sensing range and decoding range can only sense the messages sent by the originating node, but cannot decode it correctly.

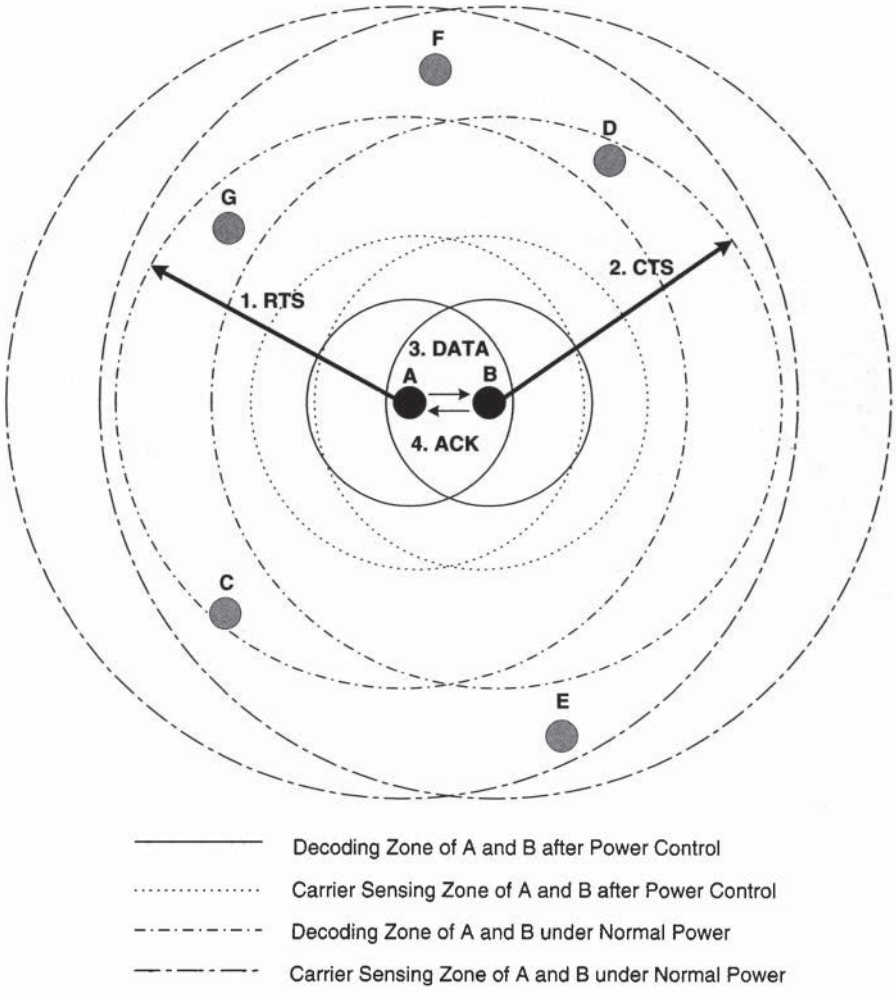


Figure 18.4 Asymmetrical link problem 2.

ent for different mobile terminals due to reasons such as interferences, distances, etc. The mix of minimum energy cost and maximum battery capacity approach generally use a threshold value to determine their favor in minimum energy path or maximum battery capacity path. Generally speaking, when the battery capacity is high, the minimum energy path should be considered. When the battery capacity is low, the maximum battery capacity path should be considered [247]. These considerations will be discussed again in Chapter 19.

18.3.2 Power Conservation Schemes under Idle Period

In an ad hoc network, not all the mobile terminals are in active communications. Some of them are idle most of the time. It is obviously wasting power for those devices to state awake instead of sleeping. As a result, power conservation schemes under idle period usually consider the suspension of some of the mobile terminals in the network [247]. However, additional time and energy are required to bring devices from sleep mode into active mode. These may affect the throughput, delay or may even increase the overall power consumption if the schemes used are not effective enough.

18.3.2.1 Sleep Mode Scheduling Although bringing the mobile terminals into sleep mode can effectively reduce the power consumption of the terminals, longer delay, lower throughput and even waste of power maybe resulted. It is because in order to deliver a message from the source terminal to destination terminal nearby, both terminals have to be awake at the same time; otherwise, the message has to be put into the buffer of the source and wait until the destination terminal awake. Consequently, long delay may result due to the waiting for the waking up of the destination terminals. At the same time, the source terminals' buffer queues may be filled up with the messages to destination nodes. Thus, packet dropping may occur during the waiting. Apart from these adverse effects, waste of power will also occur when the destination node awakes but finds that there is no message destined for it. In summary, a good sleep mode scheduling has to be used to coordinate suspension of the nodes in the network.

A straightforward approach to solve the above problems is that every node in the network will be suspended when there is no packet originated from them or destined to them. They will then wake up periodically to check for any packets. This simple idea is generally adapted in most of the approaches. Figure 18.5 illustrates the Power Saving Mode (PSM) of IEEE 802.11 in ad hoc network environment which assumes the existence of synchronized clocks between nearby mobile terminals. With this facility, the mobile terminals will wake up at the same time which is the start of the beacon interval as shown in Figure 18.5. The time period immediately after wake up is called the Ad Hoc Traffic Indication Message (ATIM) period. During this period, a mobile terminal which has packets for another nearby terminal will make an announcement during this period. As shown in Figure 18.5, mobile terminal A

has a data packet for mobile terminal B. Now, mobile terminal A will then announce this in the ATIM window. In receipt of the ATIM, mobile terminal will reply with ACK. After that, those mobile terminals which do not have any packets to send nor being announced will again change back into sleep mode. Those mobile terminals which have made announcements or being announced will stay awake for the whole beacon interval period. After the ATIM period, those terminals which have made announcements can then send a packet or several packets to the announced mobile terminals [247, 178].

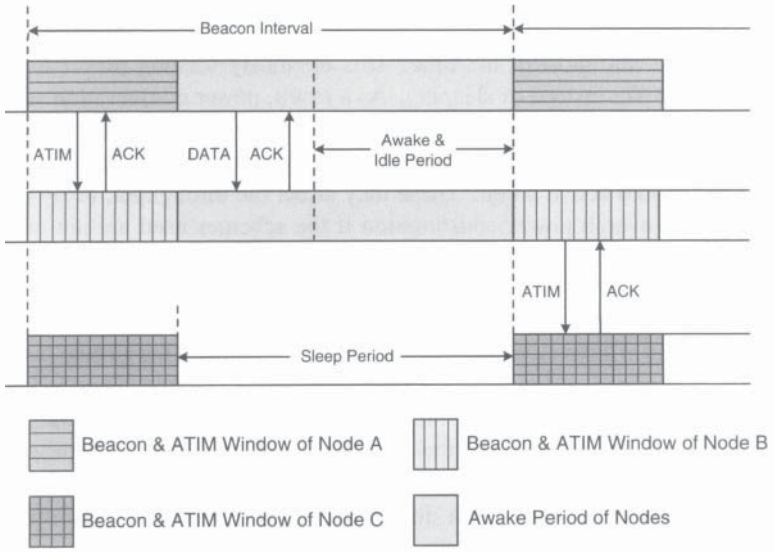


Figure 18.5 IEEE 802.11 PSM mode in ad hoc networks.

The solution suggested by IEEE 802.11 in ad hoc networking environments is based on a fixed ATIM window size for all the nodes in the network. However, some researchers have pointed out that a fixed ATIM window size does not fit all the situations well. It is because if the size of the window is too small, the nodes may not be able to make announcement for all the packets. However, if the window size is too large, more power will be consumed as those nodes are required to be awaked during the ATIM periods and also remaining time for transmitting data in the beacon intervals will be minimal. As a result, low traffic load may favor the use of a small ATIM window and heavy traffic load may favor the use of a large ATIM window.

Consequently, Dynamic Power Saving Mode (DPSM) [178] is suggested. As shown in Figure 18.6, the ATIM window of different nodes are different and will be dynamically adjusted based on the traffic loads in the network. There are two common conditions in which the window size will be increased. The first is that

there are remaining packets in the buffer which have not been announced during the original ATIM windows. The second is that a marked packet is received. As shown in Figure 18.6, mobile terminal A tries sending ATIM to mobile terminal B during the first beacon interval. However, mobile terminal B cannot receive it due to short ATIM window size. This packet is then marked. In the second trial, mobile terminal A tries sending this ATIM in the original ATIM window size. In this case, mobile terminal B receives it and finds that the packet has been marked. Mobile terminal B will then increase its window size as it finds that its window size is too small to cope with the network traffic. If all the packets in the buffer can be announced within the same beacon interval, the ATIM window will be decreased in the next interval.

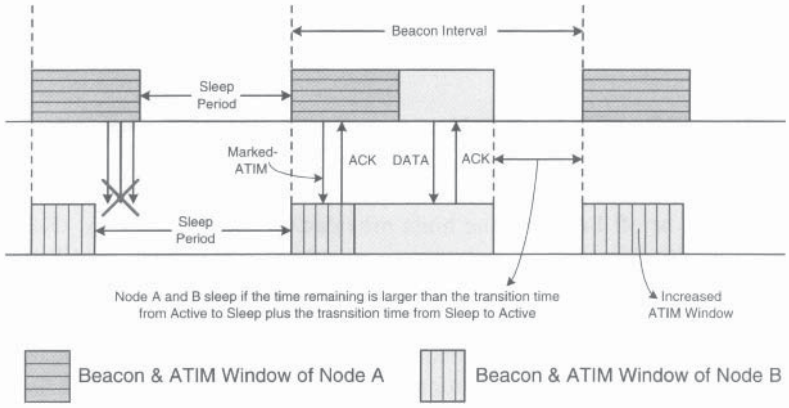


Figure 18.6 IEEE 802.11 DPSM with dynamic ATIM.

Other than an ad hoc networking environment, power saving mode (PSM) is also specified for mobile terminals in infrastructure network. However, there are some differences between PSM in infrastructure network and infrastructureless network. In infrastructure network, the access point will coordinate the access of all the mobile terminals. The mobile terminal will send out PS-Poll to the access point to see if any packet is destined for it. It will change immediately back into sleep mode if there are no pending operations (sending or receiving) that required its attention. However, this power saving mode leads to some problems as shown in Figure 18.7 for some applications. As shown in the figure, the mobile client initiates the first RPC request during the middle of the second beacon interval. It will switch to sleep mode immediately after that. In this case, the first RPC response that arrives during the same beacon interval cannot be sent to the mobile terminal immediately and the access point has to wait for the next beacon interval to do so.

In this kind of applications, power management, paradoxically, leads to higher power consumption and longer delay. This in returns lead to less work to be accomplished before the battery lifetime ends. In view of this, Self Tuning Power Management (STPM) [16] is proposed in which power management will be decided

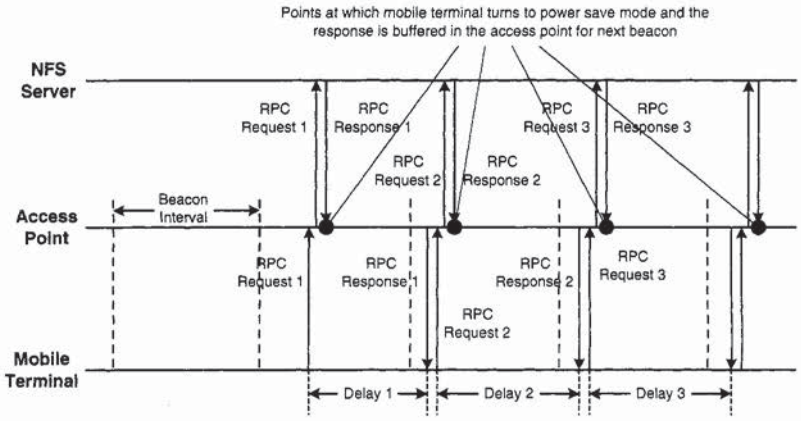


Figure 18.7 IEEE 802.11 PSM mode problem in Infrastructure network.

to be turned on or off based on the hints provided by the applications. For some delay intolerant applications (e.g., interactive applications) and large TCP transfer applications, it may be better to turn off power management.

The above algorithms assumed that all the neighbor nodes are equipped with synchronized clocks. However, this may not be the case in a mobile ad hoc network due to the mobility of the mobile terminals. Since they may not wake up at the same time due to clock skew, they have to stay awake for a longer period of time in order to increase the chance of having overlapping wake up period with their neighbors. However, even if they increase the wake up time, it is still possible for some mobile terminals to be able to send beacon and ATIM to their neighbors, but the reverse may not be true.

Consider IEEE 802.11 PSM in an ad hoc network environment shown in Figure 18.5. Suppose mobile terminal B's clock has a certain time lag compared with that of mobile terminal A and the lagging time is longer than that of ATIM period. In this case, mobile terminal B can never receive any beacon and ATIM from mobile terminal A, but mobile terminal A can receive mobile terminal B's beacon and ATIM. Consequently, various asynchronous approaches have been suggested to ensure at least certain length of overlapping wakeup time between neighbor mobile terminals, so that any pair of neighboring mobile terminals is possible to receive beacon and ATIM from their counterparts. The main reason for this to occur is that the beacon and ATIM period are set at the beginning of each beacon interval. In view of these, researchers have proposed to put beacon and ATIM period at the front in odd interval and put the period at the end in even interval. Furthermore, it is also proposed to put the beacon and ATIM period in both front and end of each beacon interval. This ensures that every neighbors can receive beacon and ATIM signal of the mobile terminal and vice versa [247].

The above solutions can solve the problem of asynchronous clocks, so that every neighbor should be able to receive the ATIM signal. However, the wake up times for the mobile terminals are much longer than that in the case of synchronous clock in order to increase the probability of overlapping between neighbor terminals. To shorten the wake up time, researchers have proposed several methods. Figure 18.8 illustrates the method in which every node will stay awake for the whole beacon interval after every T beacon intervals (e.g., $T = 3$ in the figure). In other intervals, the mobile terminals will just wake up for the ATIM period and will turn back into sleep mode if it has no packet to transmit or is not receiving any ATIM from other mobile terminals. The main drawback of this approach is that, the delay of the packet will be up to the period T.

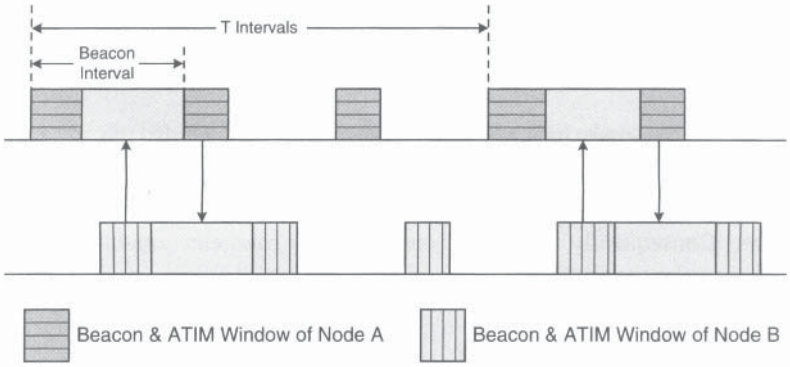


Figure 18.8 Nodes awake for a whole beacon interval every T beacon intervals for asynchronous network.

Another approach suggested by other researchers are the selection of beacon interval slots in which the mobile terminals will wake up for the whole period in the selected slots. The selection algorithm will ensure that every neighbors will be able to overlap for at least one beacon interval in those slots. Figure 18.9 illustrates the idea in which every mobile terminals will select a row and a column of slots in which they will wake up for the whole period in those selected interval slots. We can see that, suppose mobile terminal A selects the yellow row and column, while mobile terminal B selects the blue row and column, these two mobile terminals will have overlap awake period in the green slot. This slot-selection algorithm can ensure every terminals to have at least 2 slots overlap every $n \times n$ slots ($n = 6$ in the figure) [247].

The approaches of selecting rows and columns can ensure every terminals to overlap. However, a variable delay will be introduced, depending on the rows and columns selection of the two mobile terminals. In view of this, another approach is proposed as shown in Figure 18.10. Here, each mobile terminal will select a row of slots. This approach guarantees that every row will have at least one active slot overlapping with

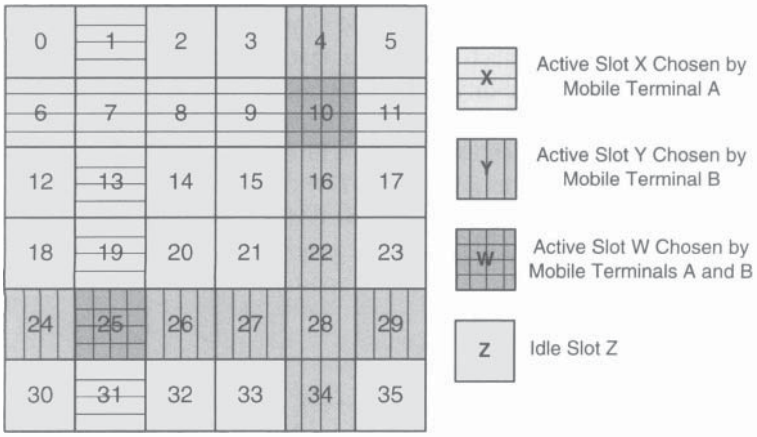


Figure 18.9 Nodes awake for a whole beacon interval during a selected row and a selected column beacon intervals for asynchronous network.

other rows. Consequently, neighbors selecting other rows can communicate during the overlap awake period [247].

18.4 PERFORMANCE ISSUES

In previous sections, various power conservation schemes have been discussed briefly. Generally speaking, it is reasonable to think that the use of power conservation schemes can save power for individual mobile terminals in the network as well as the whole network. However, various researchers have shown that the inappropriate use of the power management schemes in fact will cause throughput dropping, delay increasing, and may even cause higher power consumption due to retransmission or dropping of packets.

Moreover, researchers have found that the use of reduced power in transmission of four frames sequence will in fact cause more packet collisions. This in turns decrease the throughput of the network and increase the end-to-end delay. In view of this, a sensible approach works by using normal power to transmit RTS and CTS frames and reduced power for DATA and ACK. By doing so, the throughput is increased and end-to-end delay is reduced due to fewer packet collisions experienced. However, these two approaches still generate lower throughput and higher delay than the normal IEEE 802.11 transmission schemes without power control. However, a recently proposed scheme, called PCMAC [216] (will be discussed in next section), considers whether the transmission power will cause collision to neighbors transmission and accordingly eliminate the ACK in the sender. As a result, packet collision is reduced with the

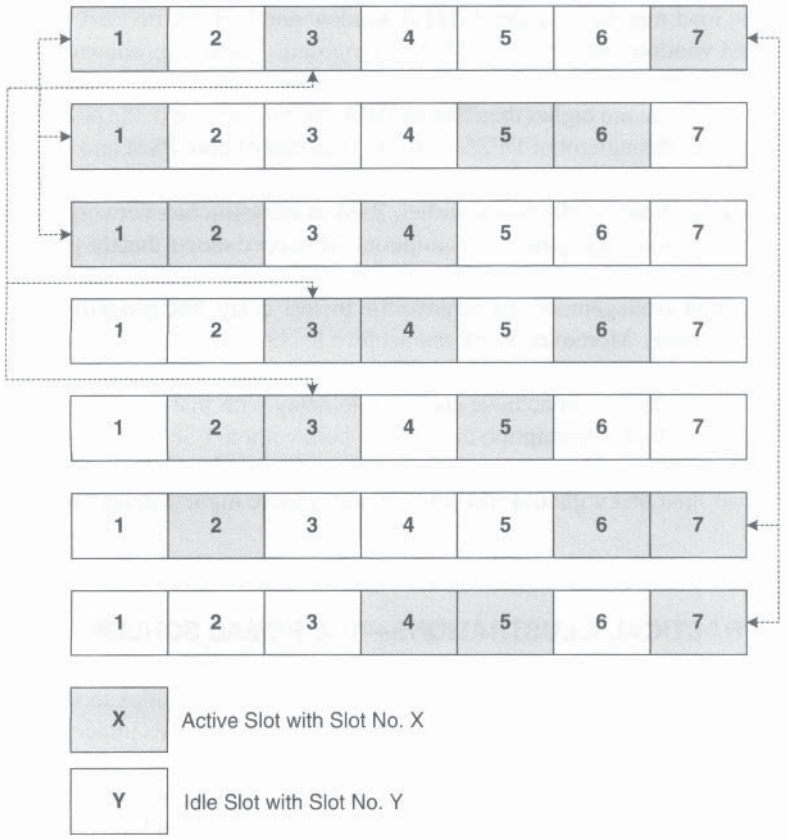


Figure 18.10 Nodes awake for a whole beacon interval during the selected beacon intervals for asynchronous network.

possibility of spatial reuse. In this case, the throughput is higher than that of normal IEEE 802.11 and the end-to-end delay is lower than that of normal IEEE 802.11 scheme without power control [216].

IEEE 802.11 has specified PSM in both infrastructure and infrastructureless networks. As discussed in previous sections, PSM in infrastructureless network makes use of the idea of ATIM window to instruct neighbors to stay awake during the beacon intervals. However, a fixed ATIM window in fact does not fit all the situations in that low traffic load may favor a small ATIM window and high traffic load may favor a high ATIM window. As a result, DPSM is proposed. The results shown that DPSM can achieve the throughput comparable with that of without PSM mode. Furthermore, both of them are higher than that of PSM. At the same time, the power used for achieving such throughput of DPSM is lower than that of both PSM and that without PSM [178].

On the other hand, as discussed earlier, PSM in infrastructure network may cause long delay and increase in power consumption. It has been shown that the performance of PSM in fact depends on applications very much. In many applications, the scheme without power management can achieve the lowest delay, but generally consumes the highest power. Moreover, PSM can achieve lower power consumption than that of without PSM. However, the delay is unacceptably high. STPM, which considers applications' features, can achieve comparable delay with that of without PSM and much smaller power consumption than that of both without PSM and with PSM. One of the interesting result is that in remote X application, PSM scheme in fact consumes more power than that without PSM scheme and achieve highest delay [16].

18.5 PRACTICAL ILLUSTRATIONS—THE PCMAC SCHEME

Various power conservation schemes have been suggested in previous sections. Power conservation schemes included both power conservation schemes under active communication and power conservation schemes under idle. Among the active power conservation schemes, power control has aroused great attention. In this section, a practical scheme known as PCMAC [216] scheme which aims at both power conservation and network throughput improvement by modifying the existing IEEE 802.11 standards will be briefly discussed. As this scheme requires only slight modifications from existing IEEE 802.11 standards, so it can be integrated into existing products easily.

In view of the asymmetric problems suggested in Figure 18.3, PCMAC is proposed to eliminate collision in both the sender (ACK from receiver) and the receiver side (DATA from sender). There are two main modifications in PCMAC. The first is the addition of power control channel with a bandwidth of 500 kbps. After the RTS/CTS exchange through the normal power, the receiver side begins receiving DATA packets. At this moment, it can estimate the noise level it can tolerate during the receipt of the

packets from the sender by:

$$\frac{P_r}{SIR_{th}} - P_n \tag{18.1}$$

Where P_r is the received signal power and P_n is the noise power perceived by receiver. SIR_{th} is the threshold signal to noise ratio at the receiver for it to correctly decodes the message.

This information will then be broadcast via the power control channel at normal power level with the frame structure shown in Figure 18.11. With the help of this information, the neighbor nodes can then estimate whether their transmissions will cause corruption in the current transmission of that particular receiver before initiating transmission.

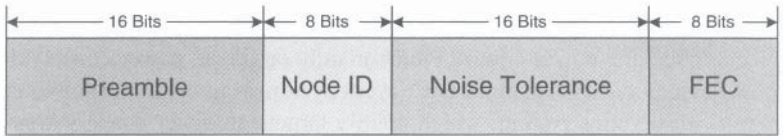


Figure 18.11 Frame structure of power control packet.

The second modification that PCMAC made is the change of four time frame sequence to three time frame sequence in which the ACK frame from the receiver is eliminated. As a result, the ACK frame will not be corrupted at the sender side. In order to facilitate the change, sent-table and received-table will be kept at sender and receiver side, respectively. The sent-table recorded the packets (including the session ID, packet sequence number and a copy of that packet) that have been transmitted to the receiver, while the received-table kept the similar items. Suppose node A is the sender and node B is the receiver. When node B received node A's RTS frame, it will reply with CTS frame with the session ID and packet sequence number of the previously correctly received packet from A. Upon receipt of CTS frame, node A will retransmit previous packet if the session ID and sequence number does not match; otherwise, a new packet will be sent.

With the help of these two modifications, packet collision can be greatly reduced. This is because the sender has to consider whether its transmission will cause collision in its neighbors by using the tolerable noise power level provided by the broadcast message of the neighbors in the power control channel. In this manner, the DATA frame in the receiver side can be efficiently protected. On the other hand, the proposed three frame sequence eliminates the ACK frame. In this case, the ACK collision at the sender side can be efficiently reduced.

18.6 SUMMARY

Since most of the mobile terminals are battery-powered, the lifetime of the devices in a wireless network is constrained by the relatively small amount of energy available. In order to extend the lifetime of individual devices which in turn leads to extension of the lifetime of the whole network, a number of power conservation schemes are proposed. At the beginning of this chapter, the characteristics of power consumption in using the mobile terminals were discussed. Generally, different amounts of power will be consumed in different states. If the mobile terminals need to communicate with other terminals in the network, different considerations at different protocol layers (Physical, MAC and Network) will also cause different power consumptions of that device and other devices in the network.

In view of these power consumption characteristics, various power conservation schemes have been proposed targeting at extending the lifetime of the devices during active communication and extending the lifetime of the devices during idle time. The power conservation schemes targeted at active communication mainly exploits the power consumption pattern in various communication layers. These lead to the use of techniques like power control which mainly targets at power conservation of individual devices in Physical and MAC layers by adjusting the transmit power level and energy-conservation routing which mainly targets at power conservation of all the devices in the network by selecting those intermediate nodes which consume the least power, which have the most abundant remaining power, etc., in participation of routing.

On the other hand, schemes targeted at idle time mainly exploit the different power consumptions of the mobile terminals in different states by putting those devices into low-power consumption states if they are not in active communication in near future. However, extra delay and power will be experienced by changing the devices from low-power state back to active state. In order for the neighboring devices to have ample opportunities to communicate with each other without unexpected and unacceptable delays, scheduling between the power-state traditions have to be considered. Generally, if the mobile terminals are connected to infrastructure or the mobile terminals are near to each other enough (e.g., within one hop), they can be assumed to have synchronized clocks. Synchronous approaches can then be adopted and devices can be assumed to wake up at the same time within each interval. Consequently, the schemes coordinate which mobile terminals can be transited back to low-power state immediately and which mobile terminals have to state awake for certain period in the interval. However, with mobility of mobile terminals, clock synchronization assumption may not be reasonable. In this case, certain clock skew may be experienced by different devices in the network. As a result, neighbors may not wake up at the same time. Thus, asynchronous approaches are proposed to tackle these issues.

Various power conservation schemes have been discussed. Generally speaking, the use of power conservation schemes may deteriorate the performance of other metrics like throughput and delay. In some cases, the power consumption of the

power conservation schemes may be even higher than that of without using power conservation schemes due to packet collision and packet dropping.

Finally, a practical power control approach—PCMAC which aims at simple modification of IEEE 802.11 in order to solve various asymmetric links problem caused by the uncoordinated power control of neighbor nodes is discussed. This approach adopts the techniques of additional busy power channel to inform the sender about the presence of current transmission and the modification of four frame sequence into three frame sequence. These techniques effectively reduce the packet collision in the network and result in throughput improvement.

PROBLEMS

18.1 Briefly discuss how power is consumed in mobile terminals.

18.2 Suggest some power conservation schemes during active communications.

18.3 Discuss some power saving strategies during idle time.

a

This Page Intentionally Left Blank

CHAPTER 19

AD HOC ROUTING

The beaten path is the safest. (Via trita est tutissima.)

—Latin Proverb

19.1 INTRODUCTION

Equipped with powerful communication hardware, many hand-held wireless devices have the capability of forming *ad hoc* connections with nearby devices. However, without infrastructure support, network connectivity is a major resource management problem.

Interoperability:

Ad hoc routing protocols were proposed in view of the additional constraints, e.g., temporarily formation, mobility of nodes, security, energy consumption, etc., that impromptu connectivity brings to the network. There are quite a number of different scenarios which focused on different constraints, some of these application scenarios are discussed in Section 19.2.

Wireless Internet and Mobile Computing. By Yu-Kwong Ricky Kwok and Vincent K. N. Lau
Copyright © 2007 John Wiley & Sons, Inc.

In order to solve the problems induced by different scenarios, at the beginning researchers suggested the adoption of traditional routing protocols—Distance Vector and Link-State used in wired networks to mobile wireless environment. This leads to the table-driven ad hoc routing protocols in which every nodes maintain a complete view of all other nodes in the network. As a result, most table-driven routing protocols are in fact variants of these two traditional routing protocols with some extension techniques to solve certain problems caused by mobility. Notable examples include the addition of sequence numbers, restricted broadcasting by some nodes, different updates for time-triggered and event-triggered updates, the grouping of certain nodes together for management, etc. Combination of different techniques lead to different performance of these table-driven protocol. These routing protocols will be discussed in Section 19.5.1.

Subsequently, researchers started thinking about the problems suggested above in another direction. Specifically, researchers become skeptical about the necessity of having a complete view of the whole networks all the time. In particular, when link breakages or mergings are frequent due to the movement of the nodes in the network. As a result, another main category of routing protocols are suggested, namely, on-demand routing protocols. These protocols do not maintain a very up-to-date view of all destination nodes in the network. They will only try to find routes to the destinations when they have traffics to send to certain destinations. Since they do not maintain a complete view of all the destination nodes, these protocols require a route discovery process and a route maintenance process. Route discovery process involves finding potential routes to a destination when there is no route set up at the time the source node wants to send packets. Route maintenance entails the process of handling link breakages at the time the traffic flows from source to destination are on-going. Such routing protocols are based on techniques like source routing, maintenance of forward and reserve path for certain source-destination pair, etc. These routing protocols will be discussed in Section 19.5.2.

Since both table-driven routing protocols and on-demand routing protocols have their own advantages and disadvantages, hybrid approaches which combines some techniques in table-driven routing protocols and on-demand routing protocols are also proposed. These will also be discussed in Section 19.5.3.

Performance:

There are quite a number of performance metrics which can be considered for routing protocols in ad hoc network. It is because different application scenarios will lead to different importance of certain performance metrics. These scenarios are usually captured by appropriate system models as discussed in Section 19.3 and these performance metrics will be further discussed in Section 19.4. However, generally speaking, average end-to-end delay and throughput, control overhead, etc., are used as performance metrics for comparison between various routing protocols. It is widely agreed that proactive routing protocols generally have higher control overhead than reactive routing protocol, as the former require both periodic and event-triggered route maintenance actions. On the other hand, the data delay performance of proac-

tive routing protocols is usually better than that of reactive routing protocols because proactive routing protocols have up-to-date routes to all destinations while reactive routing protocols may need to look for routes at that time. In terms of throughput, proactive routing protocols can be better or worse than that of reactive routing protocols depending on different scenarios.

19.2 APPLICATION SCENARIOS

Traditionally, ad hoc network came about in response to the need of setting up a network temporarily in hostile environment, e.g., battlefields or disaster sites. Under these situations, fixed and trustworthy telecommunication infrastructure are usually unavailable or are destroyed. As a result, an infrastructureless or semi-infrastructureless set up will be useful to assist military applications or rescue services.

Currently, there are three typical kinds of proposed applications for ad hoc network [28, 216].

- Figure 19.1 illustrates a pure infrasturctureless ad hoc network. In this network, nodes connect with each other to form a temporarily wireless infrastructure without the help of fixed telecommunication infrastructure. Within the network formed, users can share data and/or perform computation cooperatively. Typical example applications are:
  - Business conference meeting: Within the meeting, users may exchange data and business card.
  - Soldiers equipped with wireless devices can communicate with their allies to report the frontline situation to the commander with the help of pure ad hoc network formed temporarily.
  - When disaster situation occurs, telecommunication infrastructure may be destroyed. The rescue personnel equipped with the devices can then go into the sites and report the situations in real time.
- Figure 19.2 illustrates the hybrid infrastructure ad hoc network. In this kind of network, fixed telecommunication infrastructures are usually present. However, these infrastructures may have some deficiencies in supporting and reaching all the users. As a result, hybrid infrastructure is proposed. In those locations where fixed infrastructure may not reach, wireless ad hoc network can be formed to extend the reachability and capability of the fixed network. The main aim of this ad hoc network is to help other nodes which are out of the coverage of fixed infrastructure to reach the infrastructure network with a minimum number of hop count. Consequently, those nodes out of coverage can enjoy services provided by infrastructure network with reasonable throughput and at a smaller cost.

Figure 19.3 illustrates an example of cooperative networks—sensor networks. Sensor networks are usually deployed in a large scale in the target area. The nodes in the network will cooperate with each other to accomplish a target application, e.g., surveillance as shown in the figure. Typically, each sensor node has the following constraints:

- Low Processing Power and Limited Memory: A sensor node is usually embedded with a low processing power CPU in order to achieve low power usage and small in size.
- Battery Powered: A sensor node is usually powered by battery. As a result, energy conversation is important in such kinds of network.

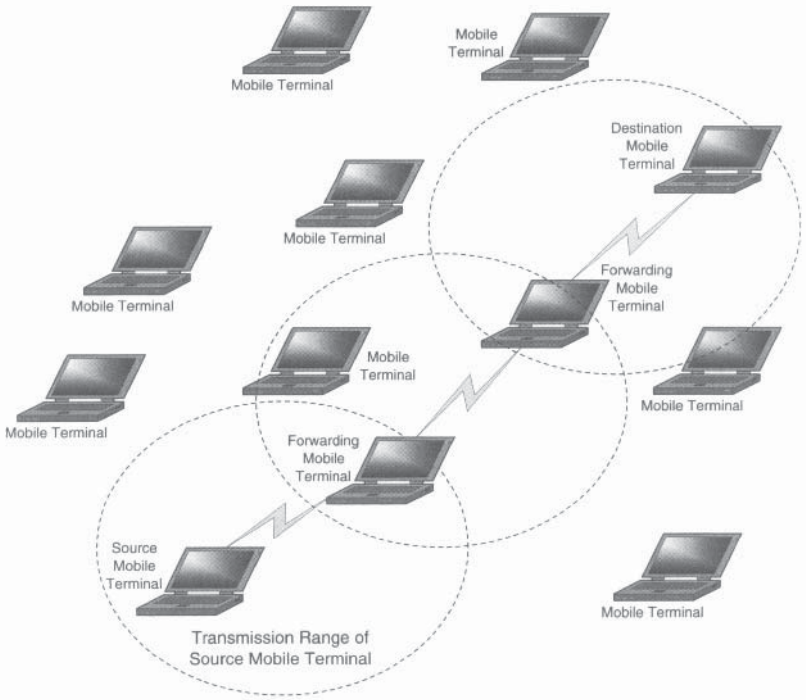


Figure 19.1 Infrastructureless ad hoc network.

19.3 SYSTEM MODEL

Section 19.2 suggested several currently proposed application scenarios, including hybrid-infrastructure, pure infrastructureless, and cooperative infrastructureless ap-

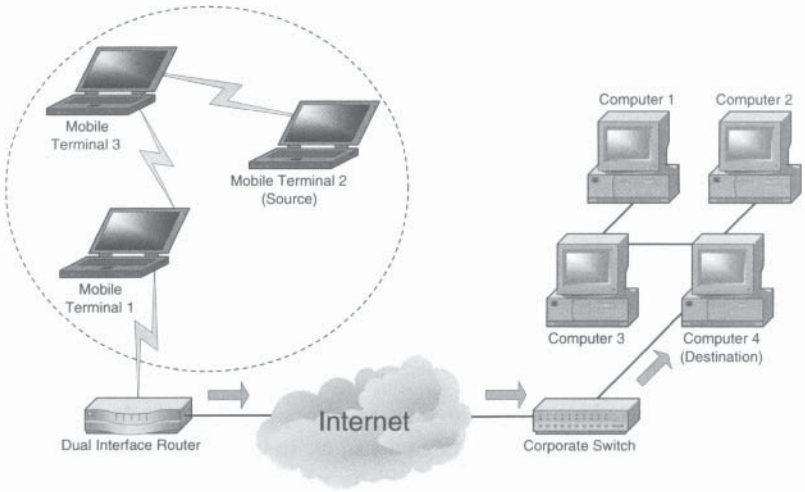


Figure 19.2 Hybrid infrastructure ad hoc network.

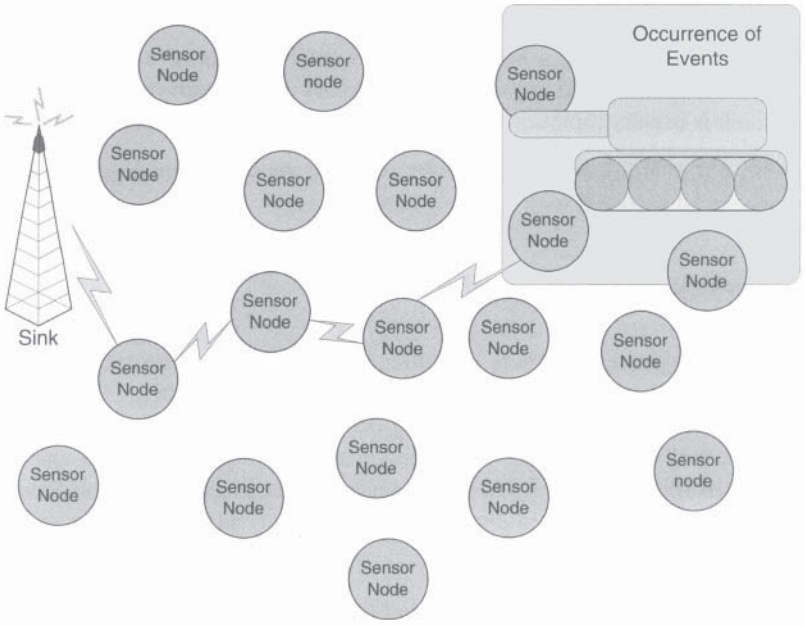


Figure 19.3 Sensor ad hoc network.

plications. However, it is difficult and expensive to deploy and perform real experiments over such a large network in a real scenario. As a result, most researchers usually performed simulations with tools such as NS-2, OPNET, etc. These simulators usually model the scenarios based on some specific system models and some of the parameters specified.

Figure 19.4 shows the system model of a commonly considered kind of ad hoc network. The parameters used depend on the application scenario. They usually include [28, 216]:

- The network is supposed to be bounded by a rectangle with size of width $\times$ height. The size of the rectangle depends on the place where the users expected the scenario to happen, e.g., 200m $\times$ 200m is usually considered a reasonable size for modeling shopping mall or exhibition centre.
- Within the boundary, there are n users equipped with mobile devices. The users may not be stationary. They may move to other location based on some mobility model with a speed denoted by vm/s. One of the commonly used model in representing the movement of users is the Random Way Point Model. Within the model, a user will choose a destination and move to that specific destination with the associated speed. After arrival, the user will pause there for some time and then move to another location following a randomly chosen direction.
- Within the network, a source user will send packets to a destination user. Such source-destination pair will be randomly chosen with variable traffic load, which is usually represented as the number of packets per second.

Based on the setting of the parameters specified above, different ad hoc network scenarios can be modeled with the use of a simulator.

19.4 PERFORMANCE METRICS

In ad hoc routing research, there are quite a large number of protocols proposed. Some of them are proposed in view of the deficiencies of others. However, the improvements are usually made by trading-off various performance considerations. It is, in fact, very difficult to get a all-round protocol which will perform the best in all of the performance metrics. However, it is generally acceptable to considered the following metrics if high speed, high throughput ad hoc network is required as used in [216].

- Average packet end-to-end delay and throughput: These two metric are generally considered as the most important metrics in many research works on routing protocols. It is because the routing protocols developed are mainly used to provide services to an upper layer network protocol, e.g., TCP. Packet

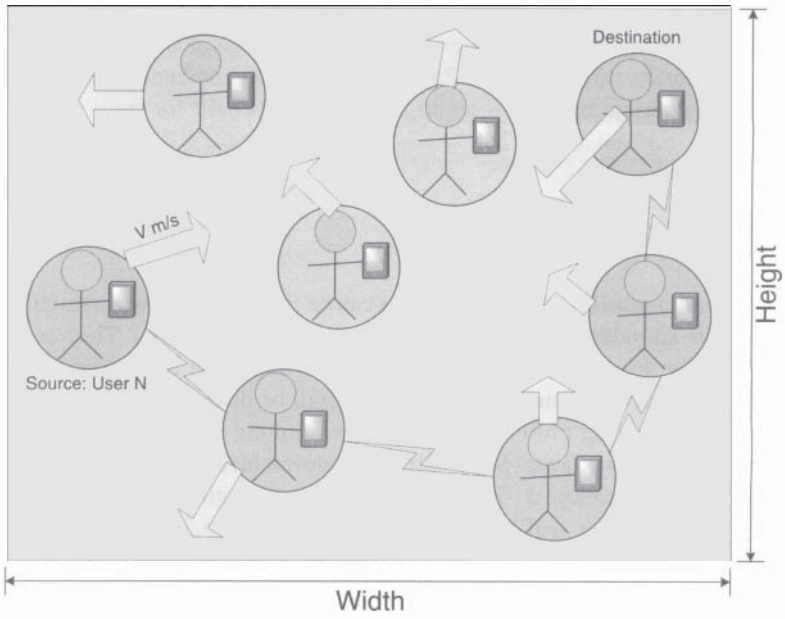


Figure 19.4 Generic system model.

end-to-end delay and throughput can be considered as the quality of services provided by the routing protocols.

- Percentage of packet delivered correctly: The wireless environment and the movement of nodes result in fluctuating wireless channel environment. This increases the difficulties of successfully delivering the packets to the target destinations. On the other hand, some applications allow certain percentage of packet errors, e.g., video streaming, which can be concealed by some error hiding and concealment techniques (this will be discussed in Section 23.4. As a result, percentage of packet delivered correctly is a good indicator on the performance of routing protocols which aim at supporting some error rate sensitive applications in upper layers.

- Control overhead: Due to different approaches on maintenance of the routing table inside the nodes, different routing protocols will induce different control overheads, which manifest as number and size of control packets, the frequency of updates, link break maintenance, etc. Generally, it is considered as unacceptable to have a large control overhead rate with respect to the data packet transmission rate.

Apart from these metrics, energy consumption is generally used as a metric in ad hoc network powered by battery. For instance, percentage of out of order packet is considered a significant metric if TCP is used. Maximum number of nodes supported will be an appropriate concern if an ad hoc network is built for supporting a large group of users (e.g., an army) [28].

19.5 ROUTING PROTOCOLS

Generally speaking, ad hoc routing protocols can be classified into two main categories based on their attitudes towards the maintenance of routing table. They are Table-Driven/Proactive Routing Protocols which will be discussed in Section 19.5.1 and On-Demand/Reactive Routing Protocols which will be discussed in Section 19.5.2[1].

As Proactive routing protocols and Reactive routing protocols have their own advantages and disadvantages. Some other protocols which would like to get the best of both worlds are developed. These protocols can generally be regarded as hybrid routing protocols which will be discussed in Section 19.5.3. Furthermore, some additional routing protocols are proposed in view of some additional constraints in the environment, e.g., Security and Energy Considerations. These will also be discussed in Section 19.5.4.

Figure 19.5 gives a brief example of some of the protocols suggested in these areas [28, 216, 140, 302].

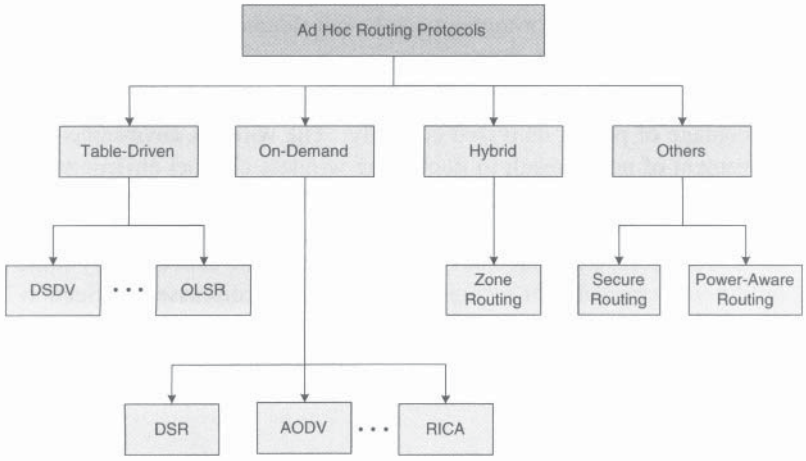


Figure 19.5 Some of the existing ad hoc routing protocols.

[1]In later sections, The terms Table-Driven and Proactive will be used interchangeably, while the terms On-Demand and Reactive will also be used interchangeably.

19.5.1 Table-Driven/Proactive Routing Protocols

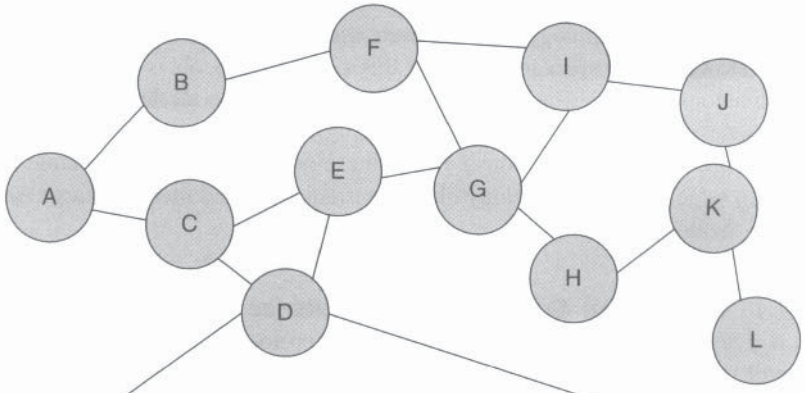
In this section, representative proactive routing protocols such as DSDV [277] and OLSR [69] are discussed. Proactive routing protocol was proposed based on some traditional routing protocols, e.g., link state and distance vector, to tackle the extra constraints induced by the ad hoc environment. The most notable constraint is the mobility of the wireless devices which can lead to link breakages frequently.

In proactive routing protocol, it is common for all the nodes to know the complete routes to all destination nodes in the network. In order to maintain a consistent, up-to-date view, all the nodes need to actively update their routing tables periodically or when the network topology changes even when there is no traffic to send in the network. However, the main advantage of this kind of protocols is that all the nodes will have a consistent view and will know suitable routes to all destinations. Thus, it is not necessary to have extra delay in finding routes to destination nodes when needed. Indeed, many people consider that these protocols are most suitable to the scenario in which network topology change infrequently and there are large traffics to send in the network.

19.5.1.1 Destination Sequenced and Distance Vector—DSDV DSDV [277] is a typical example of table-driven routing protocols. Every node in the network will maintain a routing table which has knowledge to all destination nodes in the network. Each entry in the network contains information including destination address, next hop to reach destination, destination sequence number and hop counts to destination [277, 28, 216, 140, 302]. Figure 19.6 illustrates the DSDV scenario.

Destination sequence number is added intentionally in DSDV in order to detect routing loop. It is sometimes possible for a node to learn multiple routes to a destination. For example, in Figure 19.6, it is possible for node D to receive information about getting to node B via node C and node E through completely different route, respectively. In order to decide which path to use, DSDV exploits the use of destination sequence number to resolve the problem. The one with larger destination sequence number is chosen as it represents the more up-to-update information about the destination node. However, if the two routes reaching the same destination are associated with the same destination sequence number, the one will smaller hop count will be chosen.

In order to keep the routing table up-to-date, each node in the network will broadcast their full routing table to other nodes in the network periodically with increment of their own sequence number each time. This is also known as time-triggered update. Apart from periodic update, whenever there is a change in network topology, e.g., link breakage occurs, the neighbor nodes will broadcast the event to other nodes in the network. This is known as event-triggered update. For example, suppose in Figure 19.6, the link CD breaks. Node C and node D, detecting the event, will send update packets to all the neighbors. At the same time, they will set the distance metrics associated with the broken links as ∞.



Dest	Next Hop	Dest Seq.	Hop Count
A	C	20	2
B	C	19	3
C	C	21	1
•	•	•	•
•	•	•	•
•	•	•	•

Figure 19.6 DSDV routing scenario.

In order to prevent large amount of traffics generated by the event triggered updates, only the changes from the latest time-triggered update are sent in the event-triggered update. This is known as an incremental update and is usually possible to be packaged into one packet.

19.5.1.2 Optimized Link State Routing—OLSR OLSR [69] is another typical proactive routing protocols. In a traditional link state routing protocol, every node floods the link state information to their neighbor nodes periodically. Subsequently, their neighbor nodes will then flood the information to their own neighbor nodes. This process will continue and the result is that every node in the network will have the complete view of links of all other nodes in the network. This process, in fact, causes a lot of unnecessary overhead as some of the nodes may rebroadcast information that their neighbors know already [69, 28, 216, 140].

Thus, unlike a traditional link state routing protocol, OLSR tries reducing the number of nodes which perform floodings of the link state routing protocols. Specifically, only those nodes selected by the source node (known as Multipoint Relays (MPR)) will perform the flooding actions. The neighbor nodes are selected based on the criteria that all two-hops neighbors of source node will receive the state information of source node. This is done by rebroadcasting the link state information of source node by those selected nodes inside the MPR sets. So those nodes selected as MPR will also keep a list known as MPR Selector set which stores the information about those nodes selected them as MPR, so each source node has to inform their MPRs directly about the selection. At the same time, in order to facilitate the process described above, every node in the network must have information about which nodes are within a two-hop distance. Such information is obtained by broadcasting every node's neighbor list in the beacon (or also known as HELLO message).

Figure 19.7 illustrates the idea of OLSR. In the upper part of the figure, the node E would like to broadcast its own link-state information. Its neighbor nodes C and G which are selected by node E will then rebroadcast the link-state information received by node E. After this step, in fact, all the nodes within two-hop from node E will have known the link-state information of node E as shown in the figure. In the lower part of Figure 19.7, nodes E, F, I, H are multipoint relays of node G. Upon receipt of the broadcast packet from node G for the first time, it will in turn broadcast the packet to their neighbors[2]. After this, those nodes circled in the figure will have received the link-state information of node E.

19.5.1.3 Other Protocols There are many other interesting proactive protocols: CSGR (Cluster Switched Gateway Routing) [65], WRP (Wireless Routing Protocol) [252], GSR (Global State Routing) [63], Fisheye state routing [273], TBRPF (Topology Dissemination Based on Reverse-Path Fowarding) [30], etc. Readers are referred to appropriate references for further information.

[2]Node E in this case will not rebroadcast the link-state information as it has broadcast the same information once already.

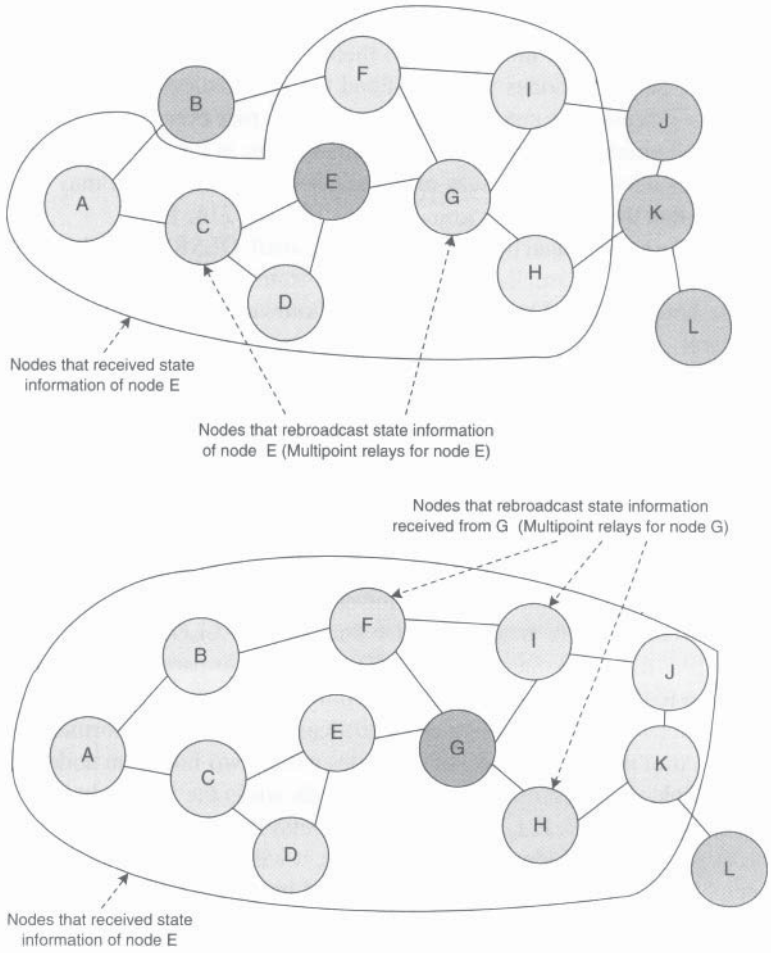


Figure 19.7 OLSR routing scenario.

19.5.2 On-Demand/Reactive Routing Protocols

In this section, representative reactive routing protocols, namely DSR [173] and AODV [278], are discussed. Reactive routing protocols were proposed in view of a high mobility of wireless devices which may result in frequent updating of routing table. These updates, in fact, are control overhead and may not be necessary in some cases, as there may not have traffic in-between these updates. To prevent this wastage of resources, reactive routing protocols are designed to try finding routes to destination only when there are traffics to destination. The main advantages are that the control overheads are expected to be smaller than that of proactive routing protocols in case of high mobility and less traffics in-between updates.

The main feature of reactive routing protocols is that they will not maintain up-to-date routes to all destinations. There are several common features among reactive routing protocols. Firstly, they usually keep a cache storing the route from a source to certain destinations that they know and maintain expiration times for the routes in order to maintain the freshness of the routes. At the same time, since there are high chances of not finding routes to destinations, they usually employ Route Discovery Process to find the route to destinations. If such destination nodes are reachable, a Route Reply will be generated back to the source with the help of intermediate nodes which will also keep table entries about these routes. In the event that certain intermediate node within a valid route is moved out of range, route maintenance process is invoked.

19.5.2.1 Dynamic Source Routing—DSR DSR [173] is a typical example of reactive routing protocols. Every nodes in the network maintains a route cache that contains the source routes to destinations that the mobile devices have learnt before. The entries inside the caches are constantly updated whenever there is more updated information passing through the node. When a source node has packets to send to a certain destination, it would first consult the route cache to see if any entry contains an unexpired route to the destination [173, 28, 216, 140, 302].

If such a route does not exist in the route cache, the source node will initiate a Route Discovery Process as shown in Figure 19.8. Here, the node S generates a RREQ (Route REQuest) packet which includes the source address, the destination address and an unique identification number, for destination D. Node S broadcasts the packet to the neighbor nodes. Upon receipt of the packet, nodes N2, N1, and N5 will check their own route caches to see if any up-to-date routes to the specified destination are known. In case that such a route does not exist, they will append their own addresses into the RREQ packet and broadcast the packet again as shown in the figure. In case that an intermediate node receives multiple broadcast messages with the same source and destination address, it will only rebroadcast the first packet it has received. As shown in the figure, the node N4 receives the RREQ packet from N1 and N6, respectively. Since node N4 received the packet from N1 first, it will only rebroadcast the packet from N1 with its own address appended.

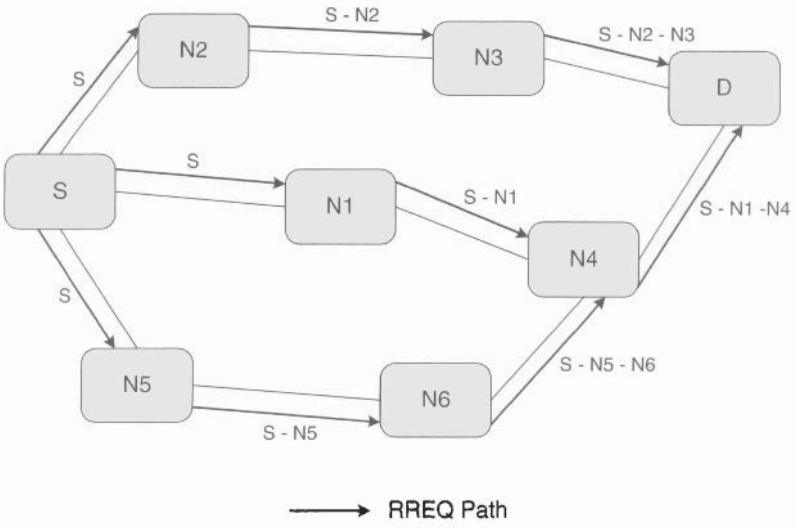


Figure 19.8 DSR route request propagation.

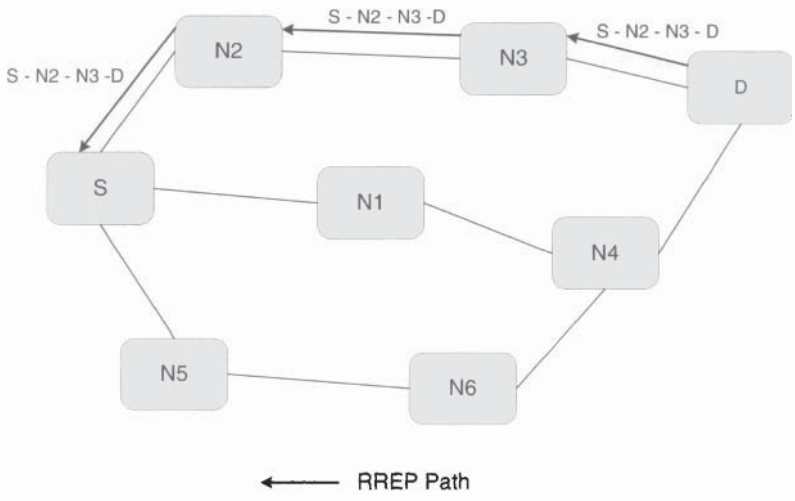


Figure 19.9 DSR route reply propagation.

When the destination node (i.e., Node D) received the RREQ packet or some intermediate node received the RREQ packet which has an up-to-date route (i.e., unexpired route) to destination node D in its route cache, the destination node or the intermediate node will generate a RREP (Route REPly) packet which contains the whole route list from source to the destination node as shown in Figure 19.9. The RREP packet will take the reverse path in the packet if the links are symmetric. The RREP packet may be routed back to source node S through another route discovery process if the links are asymmetric.

One of the unique feature of DSR is that it is a kind of source routing protocol in which every packet is included the whole route to destination inside the packet header.

19.5.2.2 Ad-hoc On-demand Distance Vector—AODV AODV [278] is another typical routing protocol of the reactive type. Its route discovery and route reply process are very similar to that of DSR. Figure 19.10 and Figure 19.11 demonstrate these two processes [278, 28, 216, 140, 302].

When a source node S wants to send a message to destination node D and does not have a valid route to that destination, it initiates a route discovery process to locate the destination by generating a RREQ packet similar to that of DSR. The RREQ packet generated contains source address, source sequence number, broadcast-id, Destination address, Destination Sequence number, and Hop count. The RREQ packet is then broadcast to the neighbors. As shown in Figure 19.10, node S broadcasts the RREP packet to its neighbors. Upon receipt of the broadcast packet, the neighbor nodes N2, N1, and N5 check to see if they have an up-to-date route to destination D. If not, they will rebroadcast the the packet to their neighbors and setup a reverse path entry inside the route table (i.e., the intermediate node will remember the first node from which it has received the RREQ packet). At the same time, the hop count field inside the packet will be incremented by one. Similar to the case in DSR, a neighbor node will only rebroadcast the RREQ packet, which is uniquely identified by the source address and broadcast-id pair which will be incremented by one every time the source address initiates a packet.

The RREP packet eventually reaches the destination node D or an intermediate node which has an up-to-date route to destination. (Whether the intermediate node has an up-to-date route or not is determined by the destination sequence number in the RREQ packet and that known by the intermediate node. Only when the destination sequence number known by intermediate node is larger than or equal to that in the RREQ packet is considered as up-to-date). The RREP packet will be route back through the reverse path setup before back to the source as shown in Figure 19.11. Upon receipt of the RREP packet, the intermediate node will also set up the forward path which points to the node where the RREP packet comes from. This route entry will be associated with a timer to maintain the freshness of the route to the destination.

However, it is still possible for multiple RREP packets to reach certain intermediate nodes or source node. In this case, instead of forwarding the packets back to the source,

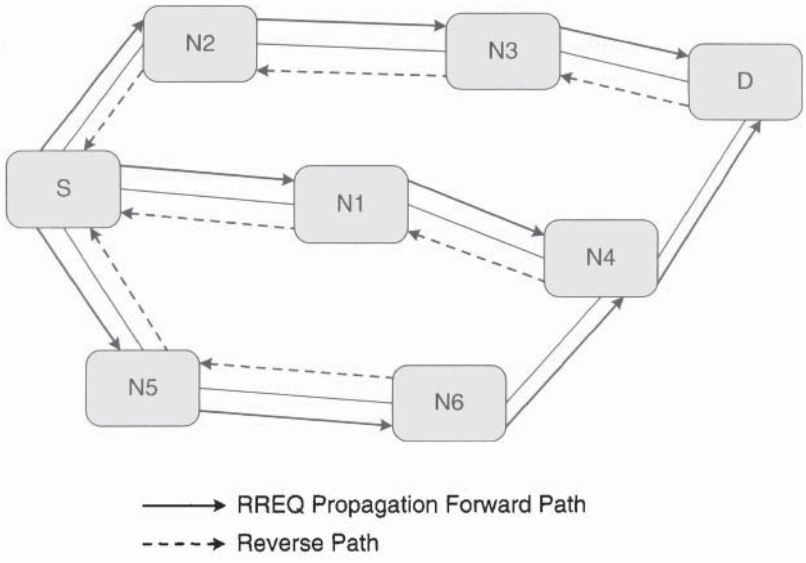


Figure 19.10 AODV route request propagation.

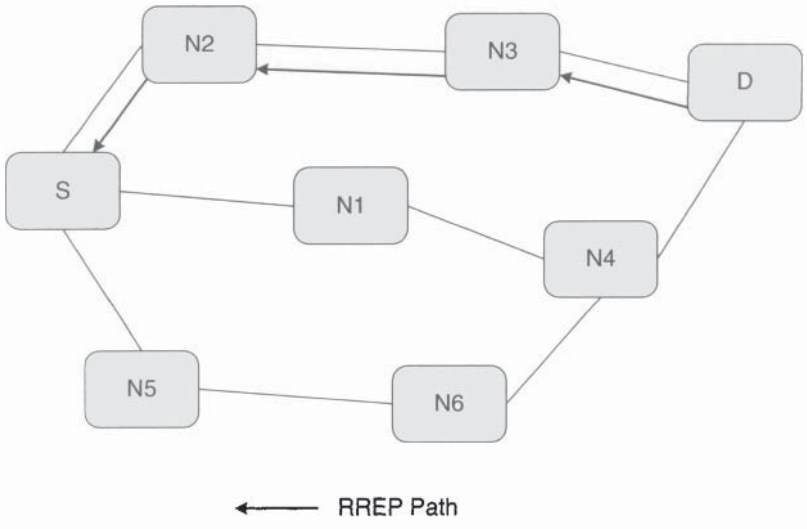


Figure 19.11 AODV route reply propagation.

the intermediate nodes will compare the destination sequence number between the one in record and that newly come RREP packet. The intermediate node will setup the forward path to the node where the newly RREP packet comes and unicast that to its reverse path neighbor only if the destination sequence number of this RREP packet is larger than that in record.

19.5.2.3 Receiver Initiated Channel Adaptive (RICA) Routing Protocol

We describe a variant of reactive routing protocols called RICA in this section. As it has been discussed, it is possible for the routing protocols to be designed to improve certain performance metrics discussed in Section 19.4. The following protocol is designed in view of possibility to improve the throughput by selecting links with high throughput to destination.

In Section 9.5, the physical characteristics of IEEE 802.11b was briefly discussed. It is known that there are 4 different transmission rate supported by IEEE 802.11b—1, 2, 5.5 and 11 Mbps. These transmission rates are supported with the use of different modulation scheme—DBPSK, DQPSK and CCK, respectively. In fact, each mobile terminal will adjust their transmission rate with different modulation scheme based on the perceived SNR or based on the CSI (Channel State Information) of the current channel.

Based on CSI, it is possible to classify each link (or hop) into different classes. In the following example, we refer to IEEE 802.11b. However, the idea can be easily extended to other IEEE 802.11 standards. Different links can be classified into different classes which can then be used to assign different weights to different links or hops. Table 19.1 gives a summary of the classification described above in IEEE 802.11b.

Table 19.1 Classification of different IEEE 802.11b physical transmission rate.

Transmission Rate (Mbps)	Modulation Scheme	Class	Weighted cost
1	DBPSK	D	11
2	DQPSK	C	5.5
5.5	CCK (4-bit)	B	2
11	CCK (8-bit)	A	1

Similar to that of DSR and AODV, RICA [216] is a reactive routing protocol which initiates a route discovery process when it requires to find route to destination. Figure 19.12 illustrates the route discovery and route reply process. At the beginning, the source node S would like to send a packet to destination D, so node S broadcasts a RREQ packet to look for the route to node D. The RREQ packet includes source address, destination address, broadcast id, hop count, CSI hop count and also lists of intermediate addresses.

Upon receipt of the RREQ packet, intermediate nodes will then:

- Append their addresses to the list of intermediate addresses.

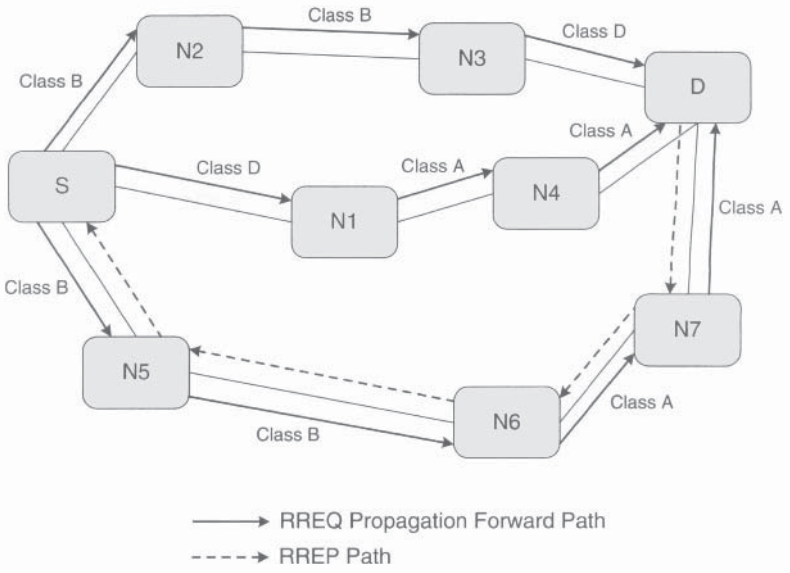


Figure 19.12 RICA route request and route reply propagation.

- Increment the hop count.
- Increment the CSI hop count by the weighted cost suggested in Table 19.1 based on the measurement of CSI of the link from where the RREQ is transmitted.

This process will continue until multiple RREQ reaches the destination node D as shown in the figure. Upon receipt of RREQ, the destination node can learn that there are several routes to destination and will try calculating the CSI hop count. In Figure 19.12, the route S→N2→N3→D has CSI hop count 15, the route S->N1->N4->D has CSI hop count 12 and the route S→N5→N6→N4→D has the CSI hop count 6. The destination node D would then generate a RREP packet which contains the source address, destination address, list of intermediate nodes, hop count, CSI hop count, reply ID, etc. This RREP packet will then be unicast back to the source. When the RREP passes through an intermediate node, a forward path to destination is also set up. At this time, node S can then decide whether to change its route or not, it will then unicast the route update packet through the selected route to the destination.

We have discussed that the channel quality between two mobile devices is time-varying. During the transmission of traffic flow from source to destination, the destination will periodically broadcast CSI-checking packet to gather information about the changing of the channel quality. Figure 19.13 illustrates the idea about the destination initiated CSI checking. The destination will send a CSI checking packet periodically to its neighbor. The packet includes similar items as that of RREQ

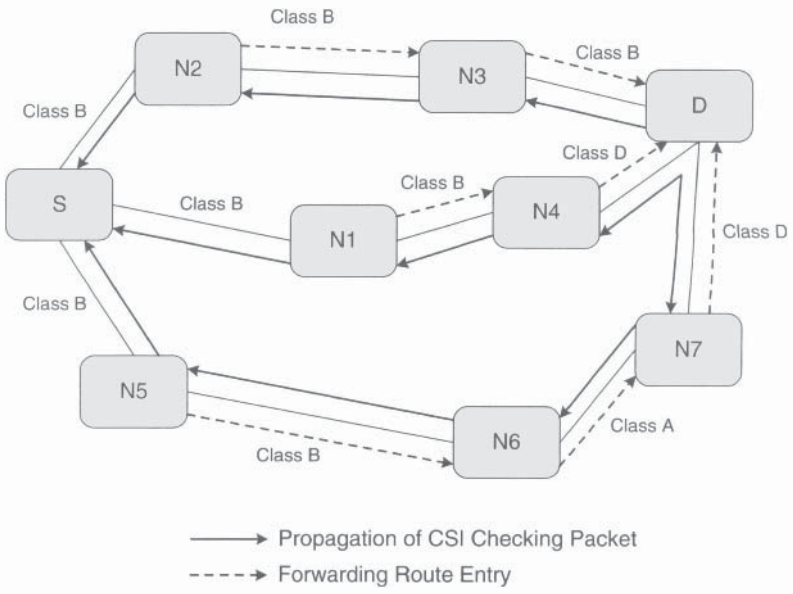


Figure 19.13 CSI checking propagation.

packet described before. However, when an intermediate node receive the packet, it will fill up the hop count field and CSI hop field similar to that suggested in RICA RREQ process. After that, it will set up a forward path to the node in which CSI checking packet arrives and rebroadcast the CSI checking packet with the updated field. In this case, similar to that of AODV and DSR, the intermediate node will only broadcast the first packet received. This process will continue until the source receives the CSI checking packet. At this point, the source can select routes again based on the shortest CSI hop to the destination by unicasting the route update back to the destination similar to that suggested above. As indicated in Figure 19.13, it is easy to see that the source node S will receive 3 copies of CSI checking packet. The source node S will know that the routes to destination via N2, N1 and N5 are of cost values 6, 15, 16, respectively.

19.5.2.4 Other Protocols There are many interesting reactive routing protocols: TORA (Temporally Ordered Routing Algorithm) [269], CBRP (cluster-based routing protocol) [170], SSA (Signal Stability-based adaptive routing) [106], ABR (Associativity-based routing) [302, 334], etc. Readers are referred to appropriate references for further information.

19.5.3 Hybrid Routing Protocols

In previous sections, two main categories of ad hoc routing protocols are briefly discussed. These two approaches have certain drawbacks, however. In table-driven routing protocol, unnecessary overheads are required periodically in order to maintain the routing table of all the nodes in the network. On the other hand, additional delay may result in looking for routes on when there are traffics. In order to get the best of both worlds, researchers proposed some hybrid approaches.

19.5.3.1 Zone Routing Protocol Zone routing protocol [272] is a typical example of hybrid approach. Generally speaking, the functions of zone routing protocol can be summarized in Figure 19.14 [28, 140].

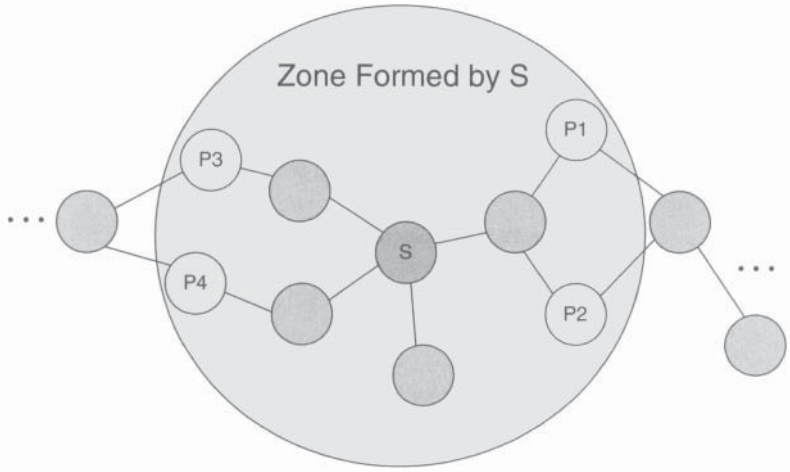


Figure 19.14 Zone routing scenario.

Under this routing protocol, each node will have a zone with certain zone radius which is defined as a predefined number of hops from the node in consideration. In the figure, the zone radius considered is two. Within the zone radius, proactive routing protocols like DSDV are used. In other words, the routing tables of every other nodes within the zone will be maintained actively. As a result, every node will know exactly how to route to other nodes within the zone radius. In Figure 19.14, node S will have direct knowledge on how to route to their neighbor nodes and border nodes, i.e., P1, P2, P3 and P4.

Suppose node S would like to send packets to a destination outside its own zone radius. In this case, node S does not have knowledge on where the destination is located. It will first send the request to their border nodes which are the nodes exactly the same number of hops with that of the zone radius, i.e., the zone radius is two and so node P1, P2, P3, P4 are border nodes in this case. Now, nodes P1, P2, P3, P4

receive the query from node S and thus, they will first look at their own route table to see whether destination node is within their zones. If not, they will continue relay the request to their border nodes. This process will continue until certain node has a route to the destination node which will then send the reply directly back to node S. As a result, this routing protocol is proactive within the zone and reactive outside the zone.

19.5.4 Ad Hoc Routing Protocols with Consideration of Other Constraints

Traditional ad hoc routing protocols suggested mainly aim at solving the problem of mobility of the nodes and temporarily formation of the network with low overhead. However, additional constraints may exist in some specific applications. Example constraints are security and energy efficiency.

19.5.4.1 Secure Routing Protocols The previously discussed ad hoc routing protocols are generally proposed with the assumption that all the nodes in the network are cooperative. However, this assumption may not be justified. In particular, the network expected to be built in hostile environment in which malicious nodes may exist or the network expected to be built temporarily and without clear identification of each nodes. Consequently, some of them may be selfish in order to save resources. There are quite a number of secure routing protocols proposed. ARAN (Authenticated Routing for Ad Hoc Networks) [309] is one of the typical example.

ARAN is developed based on an extension to AODV. It assumes the existence of a globally trusted certificate authority. Every node can get a certificate associating its public key and IP address. As shown in Figure 19.15, suppose node S would like to find the route to destination D. It will broadcast a RDP (Route Discovery Packet) packet which is a packet with packet identifier i.e., ("RDP"), IP_D, etc. The content is then digitally signed by S's private key. The RDP packet will then be sent together with $cert_S$. Upon receipt of RDP packet and $cert_S$. Neighbor nodes N2, N1 and N5 can then verify the packet by getting the S public key from $cert_S$ and decrypt the packet. After verification, reverse path is setup similar to that of AODV as shown in Figure 19.10[3]. The nodes N2, N1 and N5 will then digitally signed the RDP packet with their own private keys and again broadcast the digitally signed packet. In the next step, N3, N4, and N6 received the broadcast and perform the same actions except that they will extract the RDP packet and signed with their own signature before broadcasting. When the destination node D received the broadcast, it will then verify the source RDP packet and generate a REP (reply) packet similar to that of RDP packet except that the packet identifier is changed to REP and the IP address becomes the source IP address. The REP packet will then be forwarded back through the reverse path using similar approach to that of AODV as shown in Figure 19.11.

[3]This part is not shown in Figure 19.15

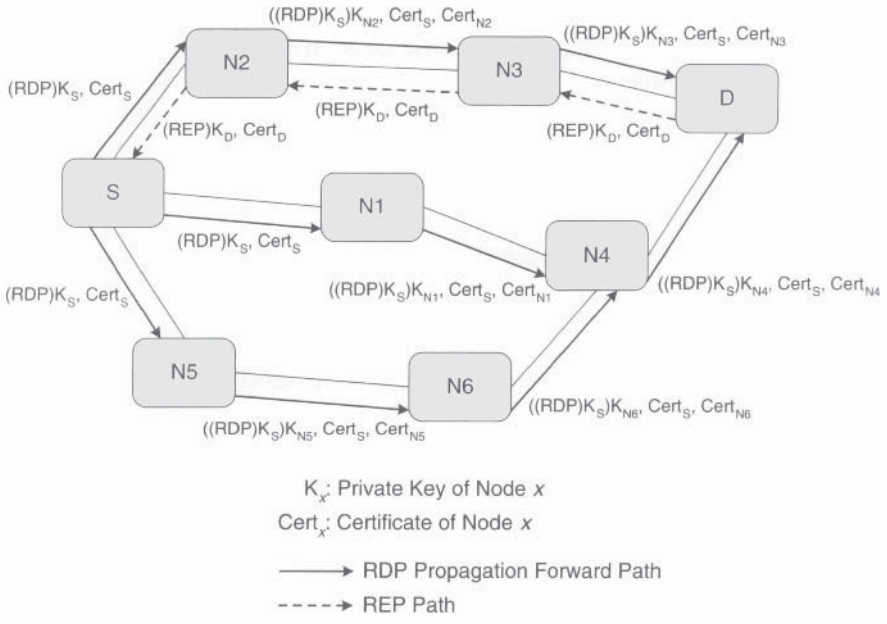


Figure 19.15 ARAN RDP and REP.

Apart from the use of certificates, there are other secure routing protocols proposed. Protocols using MAC (message authentication code) and μTESTLA to provide authenticated ad hoc routing is also suggested. Other than the use of keys and certificates, watchdog mechanisms which involve monitoring the packet forwarding behavior of neighbor nodes and pathrater mechanism which rated the path with the use of information provided by watchdog mechanism are also suggested to be extended from DSR to improve the security by selecting the route with high rate to destination [235]. CONFIDANT [43, 44] is recently suggested to further improve watchdog mechanisms and pathrater by providing trust manager and reputation system concept to evaluate the trust of neighbor nodes by combining the recommendation from other neighbor nodes observation on a certain node.

In addition to the direct security measures suggested, some multipath routing protocols are also proposed. These protocols are designed in view of the possibility of the existence of multiple disjoint routes from source to destinations. These routes can be used as a backup route in case the main route is broken or actively used to increase throughput and/or network resilient in case malicious nodes are present. Thus, the chances of packet delivery can be largely increased even malicious nodes which perform various different kinds of attacks e.g., blackhole attack, modification attack, etc.

19.5.4.2 Power-Aware Routing Protocols Traditionally, ad hoc routing is expected to solve the problem of mobility and temporary connection. Thus, intuitively, traditionally ad hoc routing protocols are expected to use the shortest path to route the packet from source to destination since minimizing the distance could lead to the highest probability of successful delivery. However, this intuition may lead to a path which uses a lot of energy. Moreover, most of the ad hoc network devices nowadays are expected to be powered by battery. The energy-deficient nodes in the network may cause the partitioning of the ad hoc network. As a result, energy consumption becomes an important consideration in ad hoc routing protocols [211].

In fact, there are quite a large number of power-aware routing protocols suggested. They can be mainly classified into two main categories—active conservation and passive conservation. There are also two kinds of objectives expected in active conservation approaches. The first kind of routing protocols aim at minimizing the power consumption of every packet sent. In this kind of routing protocol, they consider the path that can deliver the packet with minimum amount of energy from source to destination. The second kind of routing protocols aim at maximizing the network lifetime. In order words, they consider the path which may not be able to deliver the packet with minimum amount of energy, but possibly may result in maximum remaining energy in all nodes within the path after the delivery of the packet. By doing so, the network lifetime can be extended.

The passive conservation approach takes advantage of the different modes of mobile devices which have different power consumption. These modes are: transmit, receive, idle and sleep. Among them, the transmit mode requires the largest amount of energy while the sleep mode consumes the least. With the appropriate turning off the nodes in the network with minimum connectivity for routing, energy consumption in the network can be greatly reduced. For further information, readers are referred to [211].

19.5.5 Summary

In this section, we have discussed two classical classifications of ad hoc routing protocols. These routing protocols are then extended to various routing protocols with different constraints and different considerations. For instance, considering location information of nodes results in Location-Aided Routing Protocol. There are many new protocols suggested in recent years. Indeed, ad hoc routing continues to be a hot research topic.

19.6 PRACTICAL ILLUSTRATIONS—WLAN BASED AD HOC ROUTING

Various WLAN based Ad Hoc Routing protocols have been discussed in previous sections. In this section, a practical scenario of Ad Hoc Network—Mesh Network

which has been proposed recently will be discussed in detail. Some of the routing issues inside such kind of networks will also be discussed.

Figure 19.16 illustrates an example of mesh network. As shown in the figure, it is expected that within a town or a city, not all the buildings will be equipped with Internet Access capability. However, they will have WLAN facilities. In this case, ad hoc network can be formed and the buildings can cooperate with each other to route the packets from other buildings towards the location which has internet access capabilities [256].

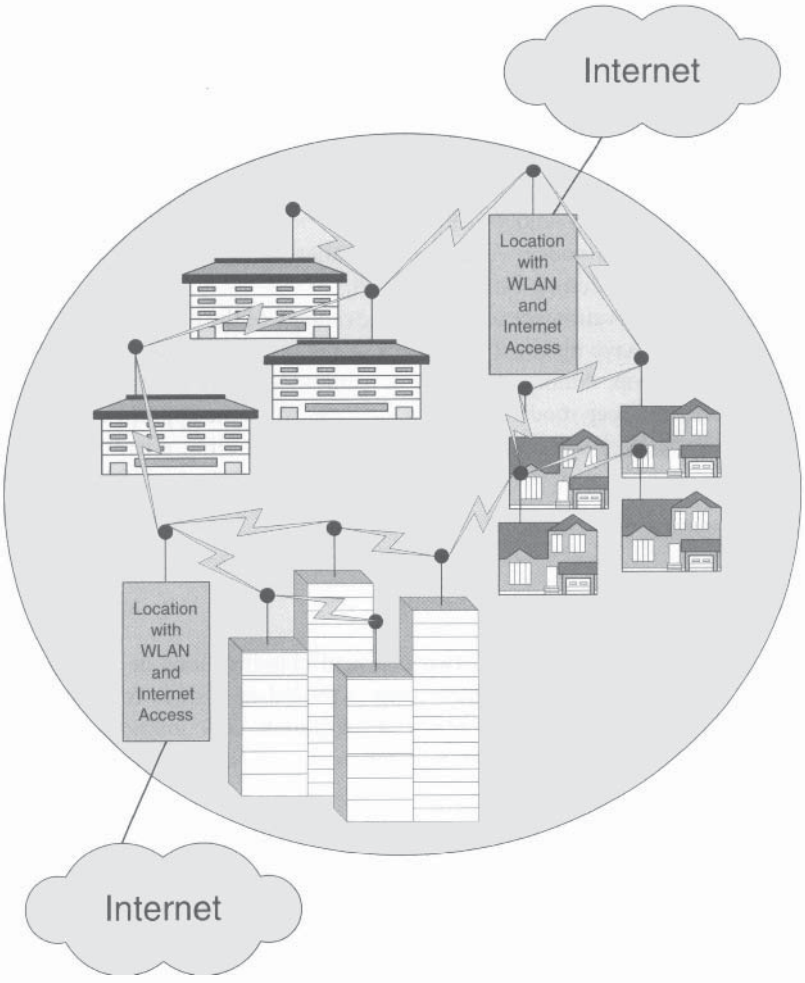


Figure 19.16 Mesh network.

This kind of ad hoc network is relatively static in the sense that the WLAN facilities are located in fixed locations within the buildings. However, it is Ad Hoc in the sense that the owners of the building may decide on their own that whether, when and how long they will turn on the WLAN facilities. As a result, it is possible for the route from source building to the location with Internet access point to be dynamic in nature. At the same time, multiple access points may exist within a town or a city. Thus, the selection of the routes may also be affected by the loads of certain access point usages at that specific period of time.

There are a number of advantages of mesh network.

- Lower cost: The cost of obtaining Internet access in such a network is expected be lower than traditional Internet access methods e.g., DSL which required every unit to subscribe their own services. With the help of mesh network, it only requires several Internet access points to exist in a town or a city. The access points can then be shared by many users through the mesh network.
- Resilient: With the help of mesh network, multi-homing is possible. As a result, the Internet access service will not be affected with the failure of single access point or single internet service provider.

Figure 19.16 shows that there are several routes from the source (e.g., residential unit) to a destination (i.e., Internet Access Point location). In this case, it is required to have an efficient routing algorithm to route the packets from source to destination. Most of the routing protocols suggested in Section 19.5 can be used efficiently here. The most commonly suggested algorithms for mesh network are DSDV, DSR and AODV. However, the main objective of these routing protocols is to find the shortest path (i.e., the smallest number of hops count path). However, it has been suggested that shortest path may not be a good path for routing the packets from source to destination. In particular, in a mesh network, throughput of the path will be more important than the shortest path. As a result, it is suggested that expected number of transmissions (ETX) [76] should be used as a metric to remedy the shortest path problem in ad hoc network routing protocols (e.g., DSDV and DSR) [76, 90]. Moreover, it is expected that the units inside mesh networks may be equipped with more than one WLAN equipment. As a result, Weighted Cumulative Expected Transmission Time (WCETT) [91] is suggested as a metric to utilize multiple channels possibility in mesh network to remedy the shortest path problem.

19.7 SUMMARY

In this chapter, ad hoc routing protocols are briefly discussed. We started our discussion by introducing some of the applications—including the usage in building pure temporarily infrastructureless network, the usage in building hybrid infrastructure and infrastructureless network in which ad hoc network can be viewed as a method to extend the capability of infrastructure network, the usage in sensor networks.

The system model of ad hoc network is briefly discussed. The model generally involves the settings such as the size of location, the number of users, the movement pattern of users, the traffic pattern, the protocols used, etc. These settings are different with different application scenarios.

There are quite a large number of routing protocols proposed in the literature and the list keeps on expanding. Generally, they can be classified into two main categories—proactive and reactive routing protocol or sometime known as table-driven and on-demand routing protocol. In proactive routing protocol, nodes will keep and maintain their routing tables (may be Distance Vector or Link State) actively and periodically. In other words, they will keep track of the routes to other destinations even when there is no traffic. These protocols are known to have relatively larger control overhead due to the periodic update which may be unnecessary or too frequent due to the movement of the nodes. On the other hand, Reactive protocols are proposed to look for the route to the destinations only when there are traffics to destination. The main drawback of these "just-in-time" protocols are that it may have long delay to look for the route. These delays may not be justified when the traffic rates are large and the mobility of the nodes are low. Due to these drawbacks, some hybrid approaches are also suggested in order to strike a trade-off between the traffic loads and mobilities. At the same time, the existence of some additional information such as location information may help in routing. For example, powered by battery or working in hostile environment, energy and security become important considerations which leads to the development of energy conservation routing and secure routing or multipath routing.

Finally, we conclude this chapter with a mesh networking scenario which is expected to be developed in coming future. This can be regarded as hybrid infrastructure and infrastructureless architecture in which residential houses are connected through wireless ad hoc network and the packets are routed to and from the access points through some ad hoc routing protocols.

PROBLEMS

19.1 Identify the reasons for the construction of ad hoc networks.

19.2 Briefly discuss the difference between table-driven and on-demand routing protocols.

19.3 In the practical illustration section, mesh network is suggested. Try suggesting some other practical scenarios in which ad hoc network routing will be useful.

CHAPTER 20

WIRELESS DATA CACHING

Few have too much and fewer too little.

—Norwegian Proverb

20.1 INTRODUCTION

Wireless data access is a typical application level activity in mobile computing. Frequently, wireless devices need to fetch information items from some servers in order to carry out its functions. For instance, a wireless PDA needs to synchronize with some data servers so as to provide updated information (e.g., news, maps, etc.) to its user.

Interoperability:

The ability to obtain on-demand information access in client-server wireless networks would provide valuable support to many interesting mobile computing applications. Researchers have proposed two promising techniques to mitigate the existing bandwidth and energy constraints: data broadcasting and client caching. Data

Wireless Internet and Mobile Computing. By Yu-Kwong Ricky Kwok and Vincent K. N. Lau
Copyright © 2007 John Wiley & Sons, Inc.

broadcasting exploits the sharing nature of the wireless medium, while client caching leverages the skewed access pattern. The major challenge of client caching is to maintain cache consistency. Clients should ensure that the cache is valid before using it to answer any query. Several cache invalidation schemes have been proposed, most of which also take advantage of data broadcasting. The basic scheme is to broadcast periodic invalidation reports (IR). Recently, the addition of updated invalidation reports (UIR) has been proposed. The IR+UIR scheme achieves a much shorter query delay than the basic IR scheme. However, existing invalidation schemes are based on two simplifying and impractical assumptions: 1.) the broadcast channel is error-free; and 2.) no other downlink traffic exists. In reality, a more realistic system model can be employed: 1.) the quality of the wireless channel is time-varying; and 2.) there are other downlink traffic sources in the system. Link adaptation allows more efficient use of bandwidth via dynamically adjusting the transmission rate with channel quality.

Performance:

The performance of various cache invalidation schemes can be evaluated based on three performance metrics: 1.) average query delay; 2.) number of uplink requests per successful query; and 3.) average delay experienced by other downlink traffic (e.g., voice). To study how the different approaches perform under various situations, we can study the performance metrics by varying system parameters like number of clients, mean query generation time, mean update arrival time, access skew (hot access probability), and mean disconnection time.

20.2 APPLICATION SCENARIOS

Scenario 1—At a Conference:

In an academic conference, researchers around the world are gathered at the exhibition center of a city. You are a speaker of a presentation session. In the last few minutes, you have updated the presentation slide, which is made available to the peer researchers via the organizer's wireless network.

You have discovered that there are slight changes in the schedule as printed on the final program booklet. You then download the updated one to your personal digital assistant (PDA) device. In addition, you discover that there is a discussion forum and go through a few interesting topics. The conference organizer also sets up a counter selling latest textbooks of your field. Sitting comfortably in the lobby, you browse the list to decide if there is any suitable one for your next semester's course.

Before departure, you want to take the chance to explore that foreign city. You query for some useful information for planning the adventure. At the end of the day, you check for the up-to-date weather information and confirm the flight schedule before going to bed.

Scenario 2—In a Shopping Mall:

In a weekend, your family is going to spend the evening in a shopping mall. Your favorite restaurant has a full house and you are put in the waiting list. The good news is that you do not need to queue up outside but can check the status via the shopping mall's wireless network. So, you query for the latest events and any special offer using your smart cellular phone. You discover that a famous boutique is having its seasonal sale and download the online catalog to have a look. Then, you decide to visit the shop and make a purchase. There is indeed a promotional campaign. With the receipt, you can participate in a lottery. After finishing the dinner at the restaurant, you query for the winner via the wireless network.

Scenario 3—In the Battlefield:

In military operations, unmanned electronic sensors are used for gathering different types of intelligent information about the enemy, such as taking photos, collecting soil samples, exploring the hostile terrain, etc. Each sensor queries the central control unit about the details of its tasks. To improve reliability, a single task may be duplicated or shared by more than one sensor. For example, sensors 1–200 are responsible for task i, which is to take photos of the missile unit, while sensors 101–200 is responsible for task j, which is to record the ambient temperature. The details of each task may change from time to time. For example, task j is changed to record the temperature only within the time period when the sensor is within a specific region. Each sensor also requests the central control unit for some common information. For example, this may include the current geographical position of itself or its peers.

Some Issues in On-Demand Information Access:

In a wireless environment, one of the distinguishing characteristics is the inherent sharing nature of the communication medium. When the server delivers a piece of information in the common broadcast channel, all the clients, within the communication range, are capable of receiving the broadcast information. Barbará and Imieliński pioneered the use of common broadcast channel to support on-demand information access [26]. This enables more efficient use of the precious wireless bandwidth.

Another promising technique is client caching. A client sends a query request to the server for a particular data item. Server replies with the requested data in the broac cast channel. In addition to answering the query, the client caches the data item in its local storage. If a new query arrives at a later time, requesting for the same data, the cached copy can be used without going through the server as before. This could save wireless bandwidth and battery energy, via reduced amount of uplink traffic. However, the use of client caching also raises another concern—cache consistency. In practice, data items are updated from time to time in an asynchronous manner. A client cannot rely on its cached copies to answer queries forever. Instead, the validity of the caches should be first ensured. The latest value model is a popular consistency model for wireless data access [48, 50, 49, 51, 156, 331], which requires that the most recent version of a data item is used to serve queries. Thus, a cache consistency scheme is needed for proper use of client caching.

20.3 SYSTEM ARCHITECTURE

Figure 20.1 depicts a typical cellular network, which consists of a number of base stations and a set of mobile users. Base stations are connected to a common database via a wired network. Each base station (server) serves the users within its covered area (cell). There are two types of users in the system, namely voice and data users. A voice session requires r_{speech} Kbps. Each data user (client) is making a file transfer session at r_{file} Kbps and also generating a stream of exponentially distributed read-only requests with mean arrival time T_q. The database is divided into hot and cold sets. Updates to each data item follow an exponential distribution with mean arrival time T_u, from which hot items have an update probability of p_u.

Figure 20.2 further illustrates the interaction between the server and the clients.

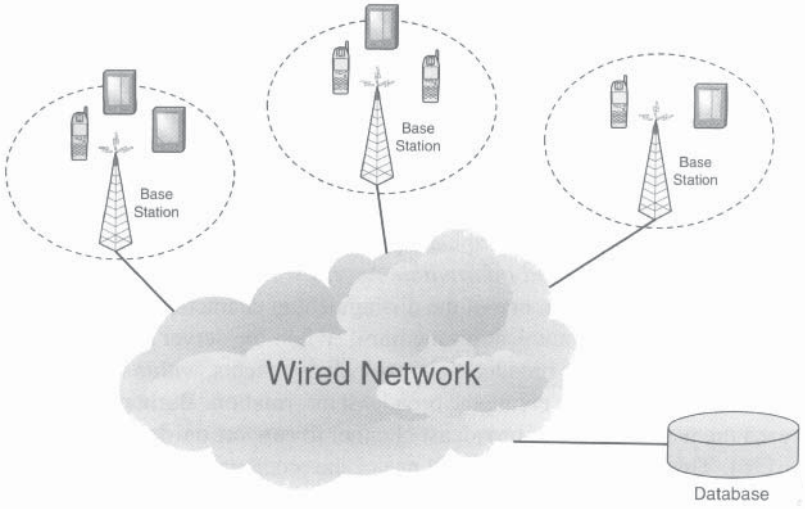


Figure 20.1 A cellular network with data and voice users.

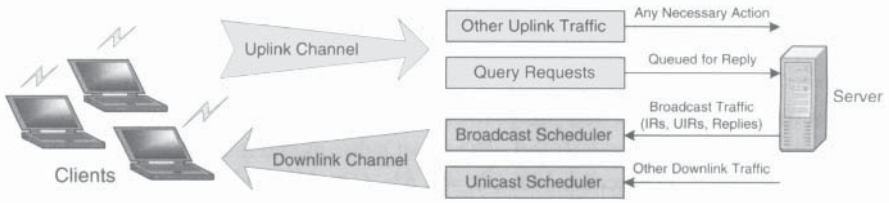


Figure 20.2 Client-server interaction.

Adaptive Physical Layer:

The wireless link between a mobile device and the base station is characterized by two signal propagation components, the fast fading component and the long-term shadowing component, which are discussed in more detail in Section 17.3. Fast fading is caused by the superposition of multipath components and is therefore fluctuating in a very fast manner (on the order of a few milliseconds). Long-term shadowing is caused by terrain configuration or obstacles and is fluctuating only in a relatively much slower manner (on the order of one to two seconds). Since mobile devices are scattered geographically across the system and are moving independently of each other, it is assumed that the channel fading experienced by each mobile device is independent of one another.

To study the channel characteristics and the effect of other downlink traffic on existing cache invalidation strategies, and to exploit the time-varying nature of the wireless channel, we can employ a variable-throughput channel-adaptive physical layer as presented in Section 17.3. The model is based on the concept of exploiting the synergy between physical layer and medium access control, instead of strictly following the traditional layered approach in protocol design.

As depicted in Figure 17.7, the receiver estimates the Channel State Information (CSI) and informs the transmitter via a low-capacity feedback channel, a facility commonly provided in practical cellular systems (e.g., cdma2000). In this system model, the base station obtains the CSI by polling a particular client via the polling subframe, and the client then responds with a known pattern in the pilot subframe (the uplink and downlink frame structure is shown in Figure 17.5). Based on the feedback CSI from the receiver, the level of redundancy and the modulation constellation applied to the information packets are adjusted accordingly by choosing a suitable transmission mode. For example, we can employ a six-mode (mode-0 to mode-5) variable-throughput adaptive bit-interleaved trellis coded modulation (ABICM) [198]. Transmission modes with normalized throughput varying from $1/2$ to $5/6$ are available depending on the channel condition.

The ABICM performs "burst-by-burst" adaptation based on the CSI measurement on the physical layer. Specifically, when the channel condition (as indicated by the CSI) is good (fading attenuation is small), the physical layer employs high order modulation and high rate error correction coding so as to boost the instantaneous throughput. On the other hand, when the channel condition is poor, the physical layer employs low-order modulation and low rate error correction coding so as to better protect the packet transmission at the expense of lower throughput. In fact, due to the salient performance in terms of system utilization, such channel adaptation techniques have been widely employed in practical systems such as 3G1x, UMTS-HSDPA, and wireless LAN systems [46].

Since the coherence time of short-term fading is around 60ms which is much longer than an information slot duration, CSI is approximately constant within a frame and it follows that the transmission mode for the whole frame is determined only by the current CSI level. Specifically, transmission mode q is chosen if the feedback CSI, $\hat{c}$, falls within the adaptation thresholds, (ξ_{q-1}, ξ_q). The adaptation thresholds are

set optimally to maintain a target transmission error level over a range of CSI values. When the channel condition is good, a higher mode could be used and the system enjoys a higher throughput. On the other hand, when the channel condition is bad, a lower mode is used to maintain an acceptable error level at the expense of a lower transmission throughput. The performance details of the variable throughput physical layer are shown in Table 20.1.

Table 20.1 Performance of the variable throughput physical layer.

Transmission Mode	Constellation	Transmission Rate (Kbps)	Target Bit Error Rate	Data Payload Capacity (bits)
0	BPSK	160	0.00002	50
1	QPSK	320	0.0001	100
2	8PSK	640	0.0005	200
3	16QAM	960	0.002	300
4	32QAM	1280	0.007	400
5	64QAM	1600	0.05	500

20.4 PERFORMANCE METRICS

The performance of various cache invalidation schemes can be evaluated based on three performance metrics:

1. average query delay;
2. number of uplink requests per successful query; and
3. average delay experienced by other downlink traffic (e.g., voice).

To study how the different approaches perform under various situations, the above performance metrics can be studied by varying six system parameters:

Number of Clients: It measures the scalability of the underlying cache invalidation scheme.

Mean Query Generation Time: It is related to client size, but queries from different clients and the corresponding replies have different corruption probability. Another difference is that larger client size means more downlink traffic in the simulation environment. Using this parameter, we would like to study how different query rates affect the performance metrics.

Mean Update Arrival Time: It aims to study different cache invalidation schemes performance under various update conditions. Update arrival time is directly

related to the number of obsolete data items. When data items are updated more frequently, cached items are more likely to be invalidated. Subsequent queries would result in cache misses.

Number of UIRs: If there are more UIRs, clients can invalidate their cache more frequently and answer queries locally with smaller delay. However, more UIRs means larger broadcast overhead, which has a negative effect on other downlink traffic. On the other hand, if the number of UIR replicates is too small, the performance gain from using UIRs is not significant. Thus, this parameter studies the optimal number of UIR replicates between successive IRs such that we can achieve performance improvement without sacrificing other downlink traffic too much.

Access Skew (Hot Access Probability): The effect of access skew on different cache invalidation schemes can be studied by varying the hot data access probability (p_q). The range of p_q value represents different query patterns. If most clients have similar query characteristics, p_q is close to 1; otherwise, pq would be just a small value. Thus, through this parameter, we can study how the performance metrics are affected under different query patterns.

Mean Disconnection Time: It studies the effect of mean disconnection time on different performance metrics. Each client has a mean connection time of, let say, 1000s. Mean disconnection period can be varied from 10s to 10^4s. Short disconnection period, e.g., 10s, is similar to hard handoff.

20.5 DATA CACHING APPROACHES

Existing cache invalidation algorithms can be broadly classified into two types: stateful server and stateless server, as illustrated in Figure 20.3. In a stateful server scheme, the server maintains a set of state information for each of the clients. On the other hand, the server is oblivious to the clients' states in a stateless server scheme. This reduces server's workload and lowers the complexity of the cache invalidation algorithm. Thus, stateless server is more popular than stateful server.

Figure 20.3 shows a simple taxonomy of a number of representative cache invalidation algorithms. The asynchronous and stateful cache invalidation algorithm (AS) belongs to the class of stateful server. Stateless server algorithms include the invalidation report algorithm (IR) and the bit sequence algorithm (BS). Due to its salient characteristics, IR is further extended with the addition of updated invalidation report, i.e., IR+UIR. IR+UIR is again further extended with some channel adaptive techniques by Yeung *et al.* in [362].

Invalidation Report (IR):

Barbará and Imieliński [26] have proposed the basic stateless server approach to maintain cache consistency. The server broadcasts an invalidation report (IR) every L

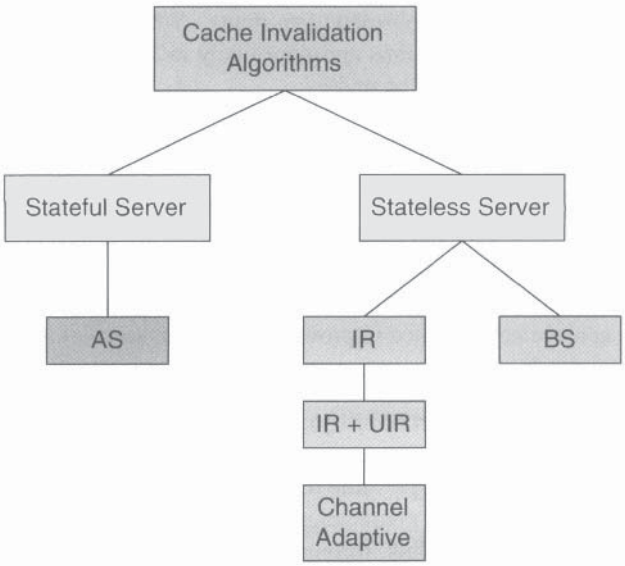


Figure 20.3 Overview of existing cache invalidation algorithms.

seconds. The primary purpose of invalidation reports is to announce a list of recently updated data items. Each client compares its caches with the list in IR and isolates any obsolete ones. Thus, the remaining copies are valid and suitable for serving future queries. Formally, each IR is a list of tuples:

$$IR_i = \{(d_x, t_x) | (T_i - \omega L) < t_x \leq T_i\}, \tag{20.1}$$

where d_x is the index of a data item x, t_x is the timestamp of its most recent update, T_i is the current timestamp, and ω is the broadcast window size, which controls the amount of invalidation information to be included in each IR. Clients do not directly use their caches to answer queries. Instead, a client waits for the next IR to determine whether its cache is valid or not. If a valid cached copy exists, the query can be served locally. Otherwise, the client issues an uplink query request to the server and waits for the reply after the next IR broadcast. As illustrated in Figure 20.4, a new query arrives at a time between T_i and T_{i+1}. If the client has a valid copy in its cache, the query can be served locally after the IR broadcast at T_{i+1}. Otherwise, an uplink request will be sent to the server. The client is expected to obtain its requested data item in the following IR broadcast, which is at T_{i+2}. Thus, the expected query delay for cache hits is $\frac{L}{2}$ while that for cache misses is $L + \frac{L}{2} = \frac{3}{2}L$. Furthermore, if a client is disconnected from the server for a period less than ωL, it can still use an IR to invalidate any obsolete data items. Thus, another function of the parameters, ω and L, is to control the longest disconnection time that can be supported by the algorithm.

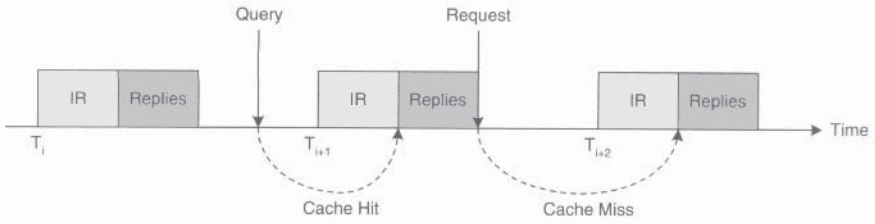


Figure 20.4 The IR cache invalidation algorithm.

Advantages of IR-based approaches include high scalability and energy efficiency: the size of each IR is independent of the number of clients; and IRs are scheduled to be broadcast periodically. As such, clients can switch to doze mode operation between successive IRs to conserve battery power. The major drawbacks, however, are: 1.) clients must flush their entire caches after long disconnection (ωL), even some of the cached items may still be valid. Since each IR contains only invalidation information in the last ωL seconds, a client cannot tell whether its cache is valid or not beyond that period; 2.) client should wait for the next IR before answering a query to ensure consistency, which potentially causes large query delay.

Invalidation Report with Updated Invalidation Report (IR+UIR):

To reduce the long query delay associated with the basic IR cache invalidation approach, Cao [51] has done some pioneering work. He proposed the addition of updated invalidation report (UIR) to the basic IR scheme (IR+UIR). Each UIR contains invalidation information for those data items that have been updated since the last IR. Formally, each UIR is given by:

$$UIR_{i,k} = \{(d_x, t_x) | T_i < t_x\}, k \in [1, m-1], \tag{20.2}$$

where $(m-1)$ is the UIR replicate times.

In other words, each UIR indicates all the updates since the last IR, which represents the most up-to-date view of the database. Clients use UIRs to invalidate their caches. If there is a valid cache, i.e., cache hit, the query can be served at an earlier time. Otherwise, i.e., cache misses, clients can also send an uplink request at an earlier time. In both cases, the query delay can be reduced.

As shown in Figure 20.5, the same query arrives at a time between T_i and T_{i+1}. For cache hit, the query can be served locally after the next UIR broadcast. Otherwise, the client issues an uplink request to the server and waits for the reply after the next IR at T_{i+1}. In this way, IR+UIR reduces the average query latency at the expense of slightly more broadcast overhead in UIRs [50, 49, 51]. Thus, the expected query delay for cache hits is $\frac{L}{2m}$ while that for cache misses is $\frac{L}{2}$.

However, the use of UIRs at a time between T_i and T_{i+1} assumes that data items in the cache are consistent up to time T_i. This condition is satisfied if the last IR is correctly received, i.e., IR at T_i. Due to disconnection problems or erroneous

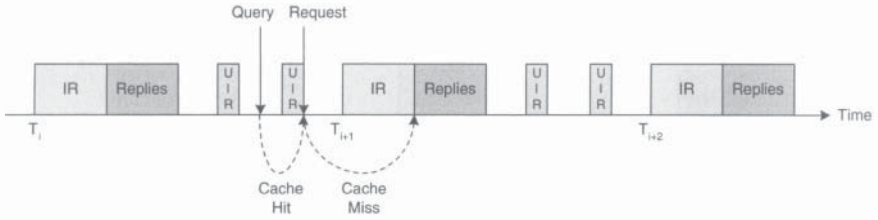


Figure 20.5 The IR+UIR cache invalidation algorithm.

transmissions, some clients may miss this last IR. These clients cannot use any of the successive UIRs before receiving the next correct IR at some later time. Equivalently, the clients are running the basic IR invalidation approach for that period. Therefore, the effectiveness of the IR+UIR scheme largely depends on the integrity of IRs.

Bit Sequence (BS):

Various approaches have been proposed to deal with long disconnection in the basic IR cache invalidation algorithm [156, 171, 331]. Jing *et al.* [171] have proposed the bit-sequence cache invalidation method (BS). Basically, each invalidation report is replaced by a set of bit-sequences. Each bit sequence records the update time of data items, which is broadcast to clients for cache invalidation. However, the BS scheme does not have the time constraint (ωL) as in IR-based schemes.

Each bit-sequence is a fixed-length bit vector. Each bit in a bit-sequence corresponds to a data item in the database. The longest bit-sequence (BS_1) is of length N bits, where N is the number of data items in the database. The bit position indicates the index of a data item in the database. For example, the n^{th} bit of BS_1 represents data item d_n. Each sequence is also associated with a timestamp, T. If the n^{th} data item is updated after T, the n^{th} bit is marked with 1. Otherwise, the bit is 0.

The second longest bit-sequence (BS_2) is of length $N/2$. The length of the next bit-sequence is $N/4$ and so on. The last bit-sequence is a two-bit vector. The k^{th} bit of these bit-sequences represent the position of the k^{th} "1" in the previous bit-sequence. If $BS_1 = 001010112$, the 1^{st} and the 2^{nd} bits of BS_2 correspond to the 3^{rd} and the 5^{th} bits of BS_1, respectively. Similarly, a value of 1 (0) represents the data item is (not) updated. If $BS_2 = 01002$, this indicates that the 5^{th} bit of BS_1 is updated, which corresponds to data item d_5. For a bit-sequence of length n, the timestamp is the time at which half of the data items have been updated, i.e., $n/2$ bits are marked 1. Initially, the number of 1's is less than $n/2$. The timestamp is then set to zero.

Compared with IR-based cached invalidation schemes (IR and IR+UIR), BS does not have the time constraint (ωL), which allows it to handle clients with arbitrarily long disconnection time. However, the underlying assumption is that the update rates of data items are not too high. Otherwise, as illustrated in the numerical example above, it is possible to result in inconsistency in client caches. Hu and Lee [156] proposed

an adaptive scheme in that the server either broadcasts invalidation reports (IRs) or bit-sequences (BSs) based on the present update rates and requests from clients. Their adaptive scheme takes advantage of the two different classes of invalidation schemes, which would perform better in a more dynamic environment.

Asynchronous and Stateful Algorithm (AS):

The previous described schemes (IR, IR+UIR and BS) are based on synchronous or periodic broadcast of different cache invalidation information: invalidation reports, updated invalidation reports or bit-sequences. These schemes assume a stateless server, which is easily scalable to a large number of clients. Kahol *et al.* [180] have proposed an asynchronous and stateful cache invalidation algorithm.

The system architecture is shown in Figure 20.6. Each mobile host (MH) is associated with a home location cache (HLC), which maintains a list of caches kept by the mobile host, i.e., the state information. If a mobile host is roaming to a foreign location, the HLC is duplicated at the foreign mobile switching station. This is to ensure that there is always an MSS (mobile switching station) to maintain the HLC for each MH at any given time. Thus, the server is stateful as opposed to the previous stateless schemes. The state information, kept at HLC, is a list of tuples: $(d_x, t_x, flag)$, where d_x is the id of the data item and t_x is the timestamp of its last invalidation. The flag is set to true when an invalidation has been sent to theMHbut no acknowledgment has been received. The distinguishing characteristic of the AS cache invalidation scheme is that the cache invalidation information is sent asynchronously and also buffered at MSS until an explicit acknowledgment is received.

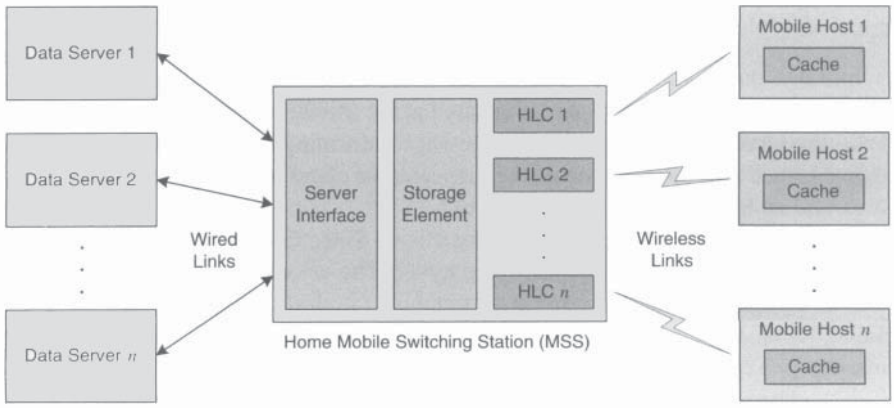


Figure 20.6 System architecture of the AS cache invalidation algorithm [180].

Each MH maintains a set of data items in its local cache. The validity of the cache should be ensured before using it to answer any query. In AS, this is achieved with the aid of the MSS. The data server informs the MSS when a data item is updated. The MSS determines, from HLCs, the set of MHs that caches the data item. Then,

the MSS sends the invalidation information to the affected MHs one-by-one. Having received the invalidation information from MSS, the MH invalidates its cached copy.

To support disconnection, each client maintains a cache timestamp, which is the timestamp of the last received message. The client includes the cache timestamp in every message to the MSS. In particular, when an MH resumes after disconnection, it sends a probe message with cache timestamp, t, to the MSS. Upon receiving the message, the MSS consults the corresponding HLC to: 1.) discard all invalidation records with timestamp less than or equal to t; and 2.) send all invalidation records with timestamp greater than t. The MH defers all queries until it receives the acknowledgment from its probe message. In this manner, an MH can determine the set of data items being updated during its disconnection.

We illustrate the operations of the AS cache invalidation algorithm with reference to the scenario shown in Figure 20.7.

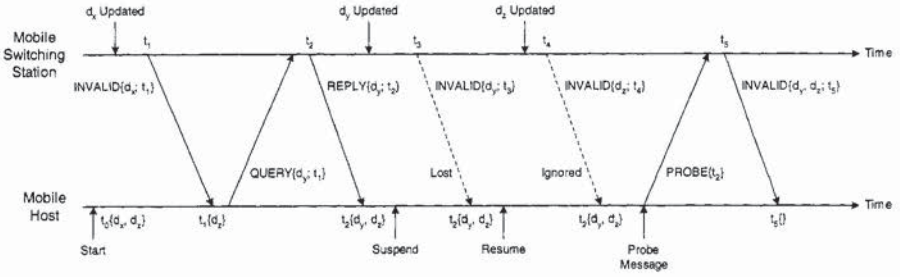


Figure 20.7 An illustration of the AS cache invalidation algorithm [180].

Initially, the mobile host (client) caches two data items, d_x and d_z, with cache timestamp t_0. The data item, d_x, is updated at the mobile switching station (server). At t_1, the server sends an invalidation message, informing the client that d_x has been updated since t_1. Having received the message, the client invalidates its cached copy of d_x and updates the cache timestamp to t_1.

A query for data item d_y arrives at the client. Since there is no cache for d_x, the client sends an uplink query request to the server. The server replies with the requested data item together with the current timestamp, t_2. Using the reply message, the client: 1.) answers the query; 2.) caches the data item d_y; and 3.) updates the cache timestamp to t_2. The client then suspends for a period of time. During its suspension, d_y is updated and the server sends an invalidation message at t_3. However, the message is lost.

Upon its resumption, d_z is also updated, the client receives an invalidation message for d_z. However, this invalidation message is ignored by the client. Instead, it sends a probe message to the server, requesting for all the invalidation records after its cache timestamp, t_2. The server consults the client's HLC and constructs the updated invalidation message. Finally, the client invalidates both dy and dz and updates its cache timestamp to t_5.

The AS cache invalidation is capable of handling arbitrarily long disconnection. However, it relies on the server to maintain a set of state information for each of the clients. Thus, scalability is a major concern. Unlike the stateless server approaches, the server conveys an invalidation message to each client in the unicast channel. If a data item, which is cached by many clients, is updated, this would suddenly generate a large amount of invalidation messages. Furthermore, the server sends invalidation messages in an asynchronous manner. This requires that a client should always in the listening state, which consumes a considerable amount of energy comparable to the receiving state [112, 111].

Reducing the Probability of Corruption in IR:

Existing cache invalidation strategies broadcast each IR in a single packet. However, this is vulnerable to transmission errors. Dividing the IR into a number of segments, with each segment separately transmitted, can reduce both the corruption probability and power consumption.

Server broadcasts invalidation reports (IRs) every L seconds. Each IR contains the current timestamp (T_i) and a list of pairs (d_x, t_x), where d_x is the id of a data item and t_x is the timestamp at which the data item is last updated. To support long disconnection period, data items that are updated in the past ωL seconds are included in each IR, i.e., $t_x > (Ti - \omega L)$, where ω is the broadcast window size. The server also broadcasts $(m-1)$ updated invalidation reports (UIRs) between successive IRs, where $(m-1)$ is the UIR replicate times. Each UIR indicates those data items that are updated since the last IR. Clients make use of IRs and UIRs to keep their caches consistent by discarding obsolete data. From the remaining valid cache, some queries can be served locally. However, the use of UIRs requires that the cache is consistent up to the last IR. If a client misses the last IR, subsequent UIRs before the next IR cannot be used. Thus, the effectiveness of UIRs depends on the integrity of each IR.

Depending on the update arrival rate to the database, an IR may include a large number of id-timestamp pairs. It is crucial to ensure that clients can successfully receive an IR. However, broadcasting the entire IR at once over the error-prone wireless channel is inefficient. This is because such a long transmission is highly vulnerable to channel errors, especially for those clients with particularly unfavorable channel conditions (e.g., with a high mobility or situates in a poor terrain). Since there may be a large number of clients listening to the same IR in the broadcast channel, it is impractical for the server to retransmit an IR to completely avoid corruption. Instead, to reduce the chance of corruption, an original IR is divided into ω segments, IR_j for j = 1 to ω. Each IR segment then contains only id-timestamp pairs in a represented broadcast period, i.e., (d_x, t_x) is included in IR_j if and only if $(T_i - jL) < t_x \leq (T_i - (j-1)L)$, for j = 1 to ω. This increases the broadcast overhead from one to ω. However, if an id-timestamp pair is included in the original IR, it appears in one and only one IR segment. The total number of id-timestamp pairs in the ω segments remains the same as that in the original IR.

Dividing an IR into ω segments can reduce both the corruption probability and power consumption. Each IR contains cache invalidation information for ωL seconds

while an IR segment contains cache invalidation information for L seconds only. Let the size of an IR and an IR segment to be $S_{\omega L}$ and S_L, respectively. Suppose the broadcast channel seen by a particular client has a bit error rate, P_e, and the overhead of each packet is x bits. In IR+UIR, the client should receive the complete IR. Thus, the probability of successfully receiving the IR is $(1 - P_e)^{(S_{\omega L}+x)}$. In the proposed scheme, the last IR segment is required. Thus, the probability of successfully receiving the last IR segment is $(1 - P_e)^{(S_L+x)}$. Since updates to a data item follow exponential distribution with mean update arrival rate of $\frac{1}{T_u}$ (recall that T_u is the mean arrival time), S_L would be smaller than $S_{\omega L}$. The success probability to receive an IR segment would be much higher than a complete IR. Thus, the client has a greater chance to benefit from subsequent UIRs. Although each IR is divided into a number of segments, the number of wake-up per broadcast period is the same as the IR+UIR algorithm. For clients with the previous IR, they are only required to retrieve the latest segment in the immediate IR. Thus, the power consumed in downloading the required cache invalidation information would be reduced.

The idea is further illustrated in Figure 20.8. In both cases, the client correctly receives the IR at T_{i-1} but some contents of IR at T_i are corrupted due to transmission errors. In the IR+UIR scheme, the client should discard the IR at T_i. There are two adverse consequences: 1.) the client cannot answer any query until the next successful IR, which is at least L seconds later; and 2.) the client cannot make use of subsequent UIRs. This increases the query delay and the client is effectively operating in the basic IR scheme. In the proposed scheme, the server divides an IR into ω segments such that each IR segment contains invalidation information for a represented broadcast interval. An id-timestamp pair appearing in the latest segment at T_i indicates the data item is updated at least once during the time period from T_{i-1} to T_i. The client can then use this latest segment, together with the IR at T_{i-1}, to reconstruct the IR at T_i. As a result, the client can use subsequent UIRs to invalidate its local cache and answer queries locally, as if there were no corrupted content in the IR.

Improving Channel Utilization:

In the wireless environment, clients experience the same broadcast channel differently [270]. Some clients perceive a good channel (e.g., the client is near to the server) while others may simultaneously perceive a poor one (e.g., far away from the server, deep fading, etc.). If the traffic is scheduled to broadcast at a fixed low rate in the hope to improve the success probability, the wireless channel is sometimes under-utilized; on the other hand, with a fixed high rate broadcast, some clients may be prone to frequent errors, which leads to increase in average query delay. Thus, dynamically adjusting the transmission rate according to the channel conditions could improve both channel utilization and overall performance.

The work in [195] is mainly about leveraging the time-varying effective channel bandwidth property to improve the performance of individual point-to-point unicast traffic. However, in the cache invalidation scenario, we are concerned about the traffic delivered over the broadcast channel. Specifically, the server cannot optimize the transmission to each particular client by using the "best" transmission mode

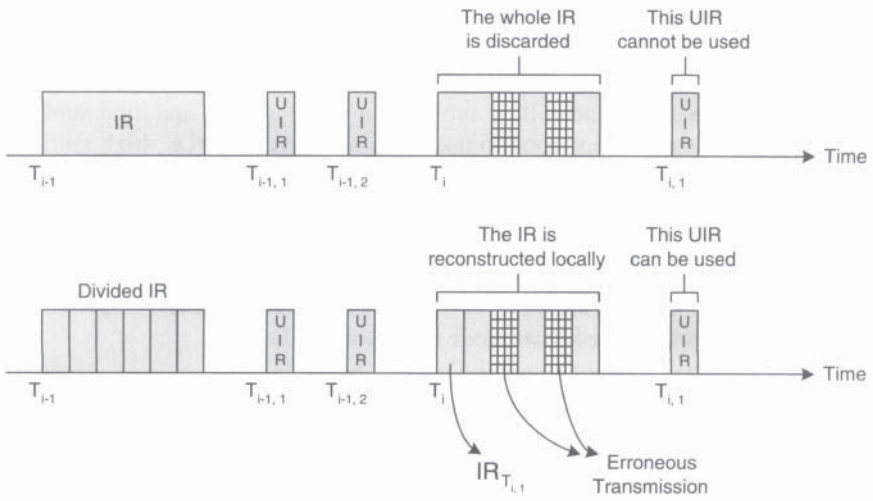


Figure 20.8 Reducing the probability of corruption in IR.

for the client. Instead, the server has to use only a single transmission mode in a broadcast message for all clients. Indeed, if a higher mode is used in the broadcast (i.e., more aggressive), some clients suffering from poor channel conditions may not successfully receive the broadcast message, despite that a shorter delay is achieved through such a high data rate transmission. On the other hand, if a lower mode is used in the broadcast (i.e., more conservative), some clients enjoying good channel conditions may not fully benefit from the good channel quality with such a low data rate transmission, resulting in a wastage of bandwidth resources.

To improve the success probability of broadcast traffic, the optimal transmission rate should depend on: 1.) the current channel status of all clients, and 2.) the importance of the information being delivered. This idea is illustrated as follows: deliver more important information using a low-rate (i.e., higher level of error protection), to improve the success probability; deliver less important information using a high-rate (i.e., lower level of error protection), to better utilize the wireless channel.

For active clients, the latest IR segments (e.g., IR_1, IR_2, IR_3), UIRs and query replies are necessary for correct operations: maintain cache consistency and serve queries. Thus, they are more appropriate for low-rate broadcast to provide better protection against transmission errors. The other IR segments, which contain old cache invalidation information, are only necessary for some clients that have been disconnected for some period of time ($< \omega L$). As such, these old segments are more suitable for high-rate broadcast to better utilize the channel. However, these low-rate and high-rate broadcasts are not kept unchanged for all IR broadcasts. What transmission rates are suitable for low-rate and high-rate broadcast is a function of

the channel conditions of all clients. Thus, a broadcast scheduler is introduced at the server side to determine the optimal transmission rates for different broadcast traffic. The broadcast scheduler collects the channel state information (CSI) from all clients with the aid of the polling subframe in the downlink and pilot subframe in the uplink. Since it is not good to use either an aggressive (i.e., high rate) or a conservative (i.e., low rate) broadcast strategy, the average data rate, computed based on all CSIs, is used to indicate the aggregate channel conditions of all clients. The average rate is used to determine the transmission mode in broadcasting the more important information. Simulation results show that the latest three IR segments are more important to mitigate the effect of transmission errors. The less important one is then transmitted with one higher mode (see Table 20.1).

Reducing Delay in Other Downlink Traffic:

In IR-based approaches, server aggregates query requests from all its clients over L seconds and broadcasts the corresponding replies after each IR. The rationale is to let more clients share the same reply via broadcast to improve efficiency, at the expense of increase in query delay among all clients. However, the channel is blocked for the whole broadcast period and other downlink traffic has to be suspended. If query replies are distributed more evenly, the impact of broadcast on other downlink traffic would be reduced.

Depending on the client size and the query generation rate of each client, there may be a long list of query replies following each IR. As discussed before, the size of each IR can be very large in order to support long disconnection period. These two factors cause the broadcast traffic to occupy the channel for a long time, resulting to a large delay in other downlink traffic. The severity of the large delay depends on the nature of other downlink traffic. For elastic traffic, there is little harm of suffering from slightly larger delay. However, large delay is unacceptable for some delay sensitive traffic such as real-time voice or video. To reduce the adverse effect of broadcast on other downlink traffic, in the proposed approach the server broadcasts query replies after both IRs and UIRs instead of just IRs. This can shorten the longest broadcast time, in turn reducing the average waiting time of other downlink traffic spent at the server.

This proposed scheme can also improve the query delay by reducing the penalty due to cache miss. As illustrated in Figure 20.9, after receiving a UIR and invalidating any obsolete data items, the client notices that a query cannot be served locally, i.e., a cache miss. The client then issues an uplink query request to the server. In the IR+UIR scheme, the query reply is scheduled to be broadcast after the next IR at T_i. Equivalently, the expected query delay due to cache miss is half the IR broadcast period, i.e., $\frac{L}{2}$ seconds. In the proposed scheme, however, the expected delay for a cache miss is the same as that for a cache hit, i.e., $\frac{L}{2m}$ seconds.

The tradeoff is the reduction in replies aggregation effect due to the shorter broadcast cycle. On the other hand, a broadcast data item is cached by all clients. When a query for the data item arrives at a later time, depending on the status of the cache, clients will have different actions. This leads to two conflicting effects: if the cached

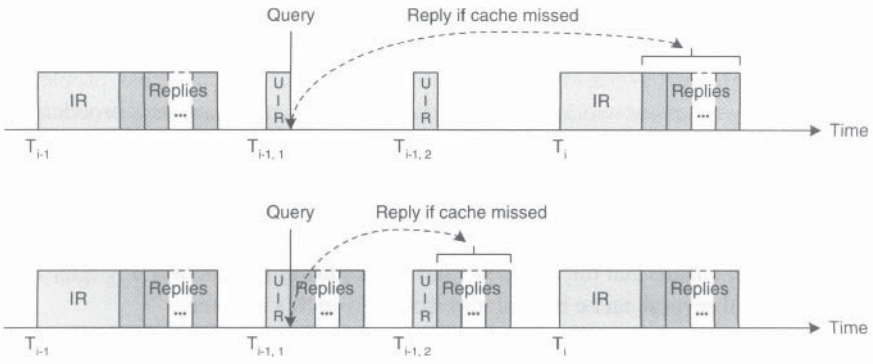


Figure 20.9 Distributing query replies after IRs and UIRs.

copy is still valid, the query can be served locally and the client can conserve battery power in uplink transmission; otherwise, an uplink request will be issued and the resources expended in previous broadcast of the data item are wasted. Which effect outweighs the other depends on the update arrival rate, query distribution in the database, and also the cache size of each client.

20.6 PRACTICAL ILLUSTRATIONS—PROACTIVE KEY CACHING

In a WLAN network, when a user wants to roam from one access point to another, it needs to go through an authentication process to perform a key exchange with the newly associated access point. This process induces a significant delay that can cause data and connections to be lost, which is particular harmful to real-time applications.

In order to solve the above-mentioned problem, Airespace has proposed a proprietary extension to the 802.11i security specification, called proactive key caching (PKC) [186]. Using PKC, when a WLAN user encounters an access point A in the first time, it is issued a key during the authentication process. When the user moves away from access point A and goes to another access point B, it will cache the key that it gets during the authentication process with access point A. Afterwards, when the user roams back to access point A, it can use the cached key to re-authenticate with access point A. This greatly reduces the authentication process. Experiments show that it can reduce 60% of time needed during the authentication process [186]. From this example, we can see that caching is a rather promising technique in wireless networks.

20.7 SUMMARY

In this chapter, we have discussed the on-demand information access problem in client-server wireless networks. Towards the end of this problem, data broadcasting and client caching are two promising techniques to mitigate the existing bandwidth and energy constraints in a wireless network. Data broadcasting exploits the sharing nature of the wireless medium, while client caching leverages the skewed access pattern. The major challenge of client caching is to maintain cache consistency. Clients should ensure that the cache is valid before using it to answer any query. We have discussed several cache invalidation schemes in this chapter.

PROBLEMS

20.1 What are the major drawbacks in invalidation based approaches?

20.2 Discuss the performance gain that we can achieve by adopting data caching in wireless networks.

20.3 What is the main issue involved in using client caching? Briefly describe how we can solve the problem.

CHAPTER 21

SECURITY ISSUES

The way to be safe is never to feel secure.

—H. B. Bohn: Handbook of Proverbs, 1855

21.1 INTRODUCTION

Providing security in wireless communication is a very difficult problem. First of all, security always involves trade-offs. Indeed, the trade-off between performance and integrity is one typical example. Users may not appreciate the high integrity provided if the performance of the communication network is considerably degraded in order to support such integrity.

Interoperability:

There are a number of problems known in current wireless security systems. Section 21.2 in this chapter first gives readers insight on general security considerations in the design of wireless communication. In order to fix some known problems, wireless systems need to make certain tradeoff between the complexity of the solutions and

Wireless Internet and Mobile Computing. By Yu-Kwong Ricky Kwok and Vincent K. N. Lau

Copyright © 2007 John Wiley & Sons, Inc.

backward compatibility. As a result, in order to remedy certain security problems, a complete upgrade in the wireless system is sometimes required. However, this is often infeasible due to the large scale deployment of the system. In viewer of these, operators or manufacturers usually prefer implementing security remedial solutions in next generation of the technologies. This results in great non-interoperable problems between different versions of the system.

As a case in point, consider that in the GSM cellular wireless system, certain flaws were found in their authentication and encryption processes. Some of them even lead to the possibility of cloning of SIM card of the phone. Certain patch-up types of solutions are incorporated in the algorithms without the need of changing of the handsets in order to provide backward compatibility. The whole authentication and encryption processes of GSM and CDMA will be summarized in Section 21.3.1 and Section 21.3.2. However, some of the possible attacks like faking of base station cannot be solved in these two second generation wireless technologies. Consequently, third generation (3G) wireless security systems are built on top of the security systems in second generation combined with fixes and improvements. Section 21.3.3 explains the authentication and encryption processes in 3G systems.

Meanwhile, Bluetooth WPAN systems come also with a quite complicated security system through a number of key generation processes to produce keys for different functions. The security system level in Bluetooth depends on the application requirements and its default level of the devices.Thus, non-interoperability may result when some of the devices do not presume adequate security support (as in the case of Internet access hotspot) or have fixed PIN code predefined. Consequently, some of the Bluetooth systems may not be able to interoperate among each other if the security levels are different. The authentication and encryption processes in Bluetooth will be discussed in Section 21.4.1, together with a recent flaw found in the pairing processes.

On the other hand, IEEE 802.11 WLANs have been deployed for years. The original design of its security system, called WEP (Wired Equivalent Security), was once considered as secure. It is because the computers at that time were not fast enough to crack such systems. Currently, a typical WEP system can be cracked very easily due to a number of flaws found in the system. In view of this, 802.11i has been proposed. In the mean time, WPA which will become part of 802.11i was released to remedy the situations. However, similar to that of cellular system, all nodes have to be WPA-compatible in order to use this new security measure. This leads to problem of interoperability to the legacy devices. At the same time, the TKIP in WPA developed based on WEP for backward compatibility is optional in the final release of 802.11i. Instead, AES will be the one adopted. However, with the use of AES, legacy devices cannot be supported. All these points will be further discussed in Section 21.4.2.

Performance:

It is notoriously difficult to quantify the performance of a security system. This is partly because a security system usually manifests as a chain of inter-related processes (e.g., key generation, encryption, etc.), among which the "weakest" link determines

the overall effectiveness of the system. Indeed, a commonly agreed paradox is that the difficulty of quantifying security undermines the security of the system.

21.2 SECURITY CONSIDERATIONS

21.2.1 Design Goals

In a wired network, malicious physical attachment to the network is difficult as it requires the attacker to intrude into the building to locate the network. However, in a wireless network, the use of a shared and open wireless medium enables a much easier intrusion opportunity. Thus, attackers have a much higher chance to successfully attack the network without being physical close to the network. There are a number of attacks on wireless system. Some of them originally were designed to attack a wired network. Nonetheless, the inherent open nature of wireless networks makes it more easier to succeed. On the other hand, some of them are specifically designed to attack a wireless system. These attacks are summarized in Table 21.1 [97, 242] below.

Table 21.1 Security threat on wireless system.

Threat Type	Description
DOS (Denial-of-Service)	This type of attack is mainly used to prevent valid users from using the network services. There are two main types—sending large amount of bogus packets to the server, e.g., TCP SYN flooding, or preventing a valid user to receive legitimate packets as the intruder performs "man-in-the-middle" attacks.
Replay Attack	The attacker gathers registration requests from a legitimate user and then replay the requests to the server so as to pretend to be a valid user.
Eavesdropping	In this type of attack, the attacker aims at uncover the secret communication data between the communication parties, e.g., username and password of network account, etc.
Session Stealing	The attacker waits for all the authentication processes to complete between the users and server, and then hijacks the session by pretending to be valid users.

In Table 21.1, a number of attacks which can be targeted on both wireless and wired network environment are listed. Since a wireless environment makes these attacks easier to be successful, new wireless security solutions are specifically designed to tackle these attacks and some other wireless specific vulnerabilities.

21.2.2 Components of a Secure Wireless Communication System

In a wireless communication system, it is necessary to ensure two main services—authentication and privacy [97, 242].

Authentication is necessary in a wireless communication system in order to ensure the correctness of identity of the communicating party. In particular, systems that bill the users are required to authenticate the users to ensure that they are valid users and charge their accounts accordingly. Generally speaking, authentication in a wireless environment requires both party to have a shared secret which will then be used to encrypt the identity proof or generate more keys for later use.

On the other hand, since a typical wireless environment is open so that eavesdropping is usually practical. As a result, privacy becomes a serious concern. It is necessary for a wireless system to ensure that the message is only readable by valid target parties, unmodified during the transmission, and remains intact even being intercepted. In a wireless environment, this is generally achieved by encryption of the data using keys generated and only known to the targets parties.

In fact, with the careful handling of authentication processes and privacy issues together with management of the keys. Most of the attacks mentioned above can be effectively reduced. In the following sections, security issues and techniques used in various wireless system will be elaborated.

21.3 CELLULAR WIRELESS SECURITY

Generally speaking, cellular operators have adopted effective measures in authentication and encryption of the messages.

21.3.1 GSM Security

During the development of GSM security techniques, it was decided to make them secret (without publishing for public reviewing). This, in fact, arouse great public and hackers attention. Finally, several flaws and problems were discovered. In the following, the authentication and encryption used are discussed [237].

The security of GSM starts from its SIM (Subscriber Identity Module) card. It is a smart card storing various important information items including IMSI (International Mobile Subscriber Identity), TMSI (Temporary Mobile Subscriber Identity), Ki, $A3/A8$ algorithm, etc. IMSI is used to uniquely identify a specific user globally and TMSI is assigned for subsequent identification within an area after IMSI is used for the first time. The use of TMSI, in fact, provides anonymity to the users as the attackers may not discover the parties under communication easily. Since a SIM card stores much sensitive information, it has to be protected for authorized access only. It is generally protected with a PIN (Personal Identity Number). After a preset number of failed attempts, the card will be locked and can only be unlocked by using the

PUK (Personal Unblocking Key). The card will be locked permanently if the PUK is wrongly input for a number of times.

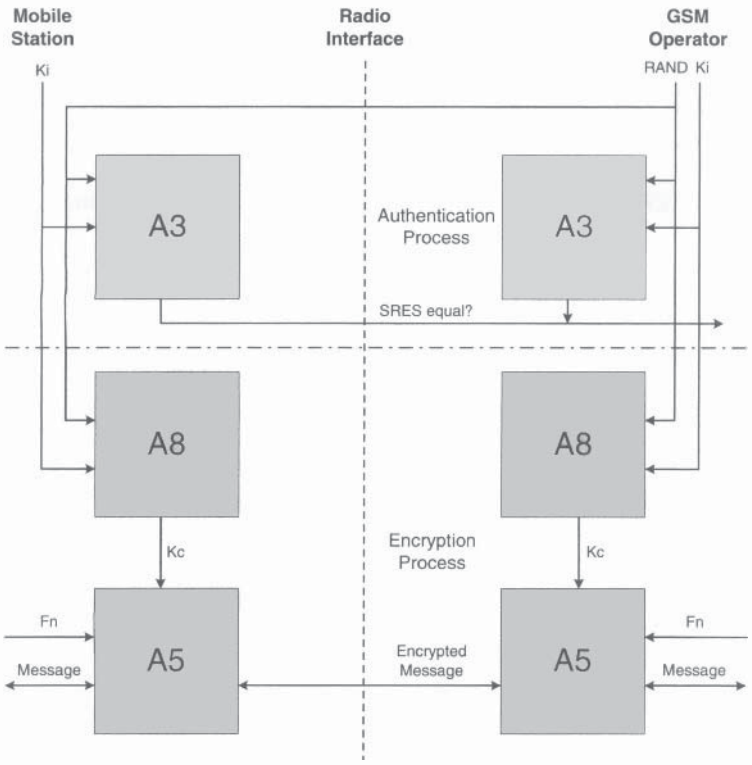


Figure 21.1 GSM authentication and encryption mechanism.

Figure 21.1 illustrates the authentication and encryption processes carried out in a GSM network. Ki in the SIM card plays an important role in the proof of identity and generation of subsequent keys. The key steps are listed as follows.

1. The base station generates a challenge (RAND) and sends it to the phone.
2. The SIM card with the valid Ki encrypts the RAND and produces a SRES (signed response).
3. At the same time, the VLR serving the mobile phone asks for a copy of A3 and Ki from the HLR which in turn obtains *a priori* the information from AuC (Authentication Center).
4. The VLR calculates the SRES from $A3$ and Ki.

5. Phone sends SRES to base station which then forwards it to VLR for comparison. The user is successfully authenticated if the SRES values calculated by both parties match. The user is then allowed to use the network.

The above processes allow a GSM network operator to efficiently authenticate the users. Originally, it was suggested that anonymity is provided with the use of TMSI. However, only anonymity is not enough. Indeed, it is also necessary to protect the user's privacy including the voice and data communication privacy. This is achieved by the encryption of voice and data messages. As shown in Figure 21.1, the encryption processes in GSM are done by the cooperation of $A8$ and $A5$ algorithms, which are briefly described below.

1. The SIM card of the mobile phone receives the RAND at the time of authentication. It will then be processed by the $A8$ algorithm together with Ki to produce the session key Kc.

2. The GSM network operator carries out the same computation to generate the Kc specific to that particular user.

3. The whole communication process can then be encrypted with the use of Kc and message in the $A5$ algorithm.

The above processes provide communication privacy to the GSM users. However, GSM authentication and encryption processes are found to be flawed. Some of the well known problems are listed below.

- Ki in a SIM card can be extracted with physical contact for minutes with the use of side channel attack. As a result, the SIM card can be cloned.

- It was claimed that $A3/A8$ algorithms are based on 64-bit keys Kc. However, the last 10 bits of the keys are fixed as 0 bits. As a result, brute force attacks become much more effective.

- $A5$ was discovered to be flawed also.

- The authentication process only involves mobile phone authenticating itself to the base station, but not vice versa. Thus, a fake base station can be used to capture secret user messages.

21.3.2 CDMA Security

In fact, CDMA is very similar to GSM in the sense that it also uses similar approaches in authentication and encryption processes. However, unlike GSM, user information is stored in the handsets instead of a SIM card. A critical information item is a 64-bit symmetric key known as A-Key [237, 291]. This key is stored into the phone and

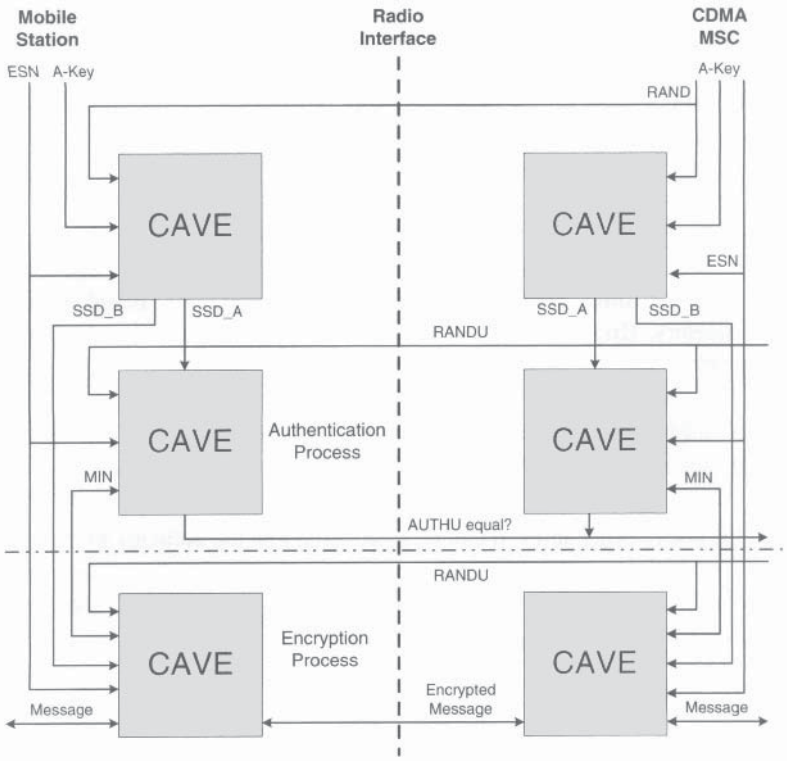


Figure 21.2 CDMA authentication and encryption mechanism.

operator database at the time of purchase. At the same time, the phone itself contains a unique ESN (Electronics Serial Number). These two information items uniquely identify a user.

CDMA authentication and encryption processes are illustrated in Figure 21.2. The authentication steps are summarized below.

1. The mobile phone gets a RAND from the operator and generates two 64-bit keys—SSD_A and SSD_B with the use of the CAVE (Cellular Authentication and Voice Encryption) algorithm, ESN, A-key and RAND. These two keys are also stored inside the database of the operator (generally inside AuC).

2. When the mobile phone indicates to the network that it would like to make a call, MSC in the operator network would try getting the user's information from the HLR and generate a RANDU (challenge) to the user.

3. With the knowledge of SSD_A, ESN, MIN and RANDU, the mobile phone is able to generate a 18-bit AUTHU which is a proof of knowledge of A-Key. At the same time, MSC will calculate the same AUTHU based on the same parameters. (In some case, MSC may not have the ability to calculate CAVE algorithm, then AuC will take over the task).

4. The mobile phone transmits the AUTHU to MSC. If the AUTHU values from both side match, the authentication is successful and the call can be proceeded.

From the above procedures, it can be discovered that the authentication processes in CDMA network is very similar to that of GSM network. The encryption processes are summarized below. (The readers are also referred to Figure 21.2 for illustration of simplified version of encryption processes.)

1. The mobile phone gets the RAND from the operator. With the knowledge of another key—SSD_B, ESN and MIN, a 18-bit VPMASK (Voice Privacy Mask) can be calculated using the same CAVE algorithm.

2. MSC calculates the same VPMASK using the same algorithm and parameters. This VPMASK can then be used to encrypt the voice communication[1].

With the use of CAVE and A-Key, a CDMA operator can provide authentication and encryption to their users. CDMA takes a similar approach as that of GSM in that it keeps the algorithm unpublished. Fortunately, there are fewer flaws found in CDMA.

[1]There are other keys generated using similar parameters for signalling message and data encryption.

21.3.3 3G Security

We have briefly outlined second generation cellular security systems. As can be seen, they have several deficiencies including poor encryption algorithm, lack of mutual authentication, difficult upgrading if flaws are discovered, etc. In view of this, 3GPP (Third Generation Partnership Project) which was developed to define standards on 3G (i.e., WCDMA) also established a task group for security [39, 237]. The task group proposed the use of AKA (Authentication and Key Agreement) protocol in 3G network in which the encryption algorithm used is Kasumi, which is widely considered to be effective.

The new authentication and encryption processes are illustrated in Figure 21.3 and the corresponding functions are summarized below.

- The main function of $f1$ is to generate an authentication token (MAC) to authenticate the network operator. This is derived based on the value of K (a root key known only to the operator and stored in the SIM card), SQN (Sequence number) and AMF (implementation specific to network operator).
- The main function of $f2$ is to generate XRES to authenticate the users to the network operator. Consequently, a mutual authentication can be carried out.
- The main function of $f3$ is to derive CK (Ciphering Key) which will be used in $f8$.
- The main function of $f4$ is to derive IK (Integrity Key) which will be used in $f9$.
- Algorithm $f5$ is used to generate AK (Authentication Key). This key is used to encrypt the sequence number to mobile station.
- Algorithm $f8$ is used for encryption of the message. It has a number of new features that are not found in second generation cellular service. This algorithm takes in 128-bit CK, COUNT-C, DIRECTION, BEARER and messages as parameters to produced encrypted ciphers. COUNT-C is mainly used to prevent replay attack and make sure the cipher is updated in every round. The DIRECTION is mainly used to prevent replay attack, while the BEARER is to ensure different values for multiplexed link of a particular user.
- Algorithm $f9$ is used for providing integrity protection to the message. This ensures the message to be unmodified. It makes use of 128-bit IK, COUNT-I (integrity sequence number), fresh and DIRECTION to hash the message. The use of COUNT-I and fresh together prevents the replay attack as COUNT-I and fresh will change frequently. On the other hand, the use of DIRECTION prevents the reflection attack.

From the above description, it can be seen that there are a number of changes made in 3G security system in order to improve security. These include the mutual

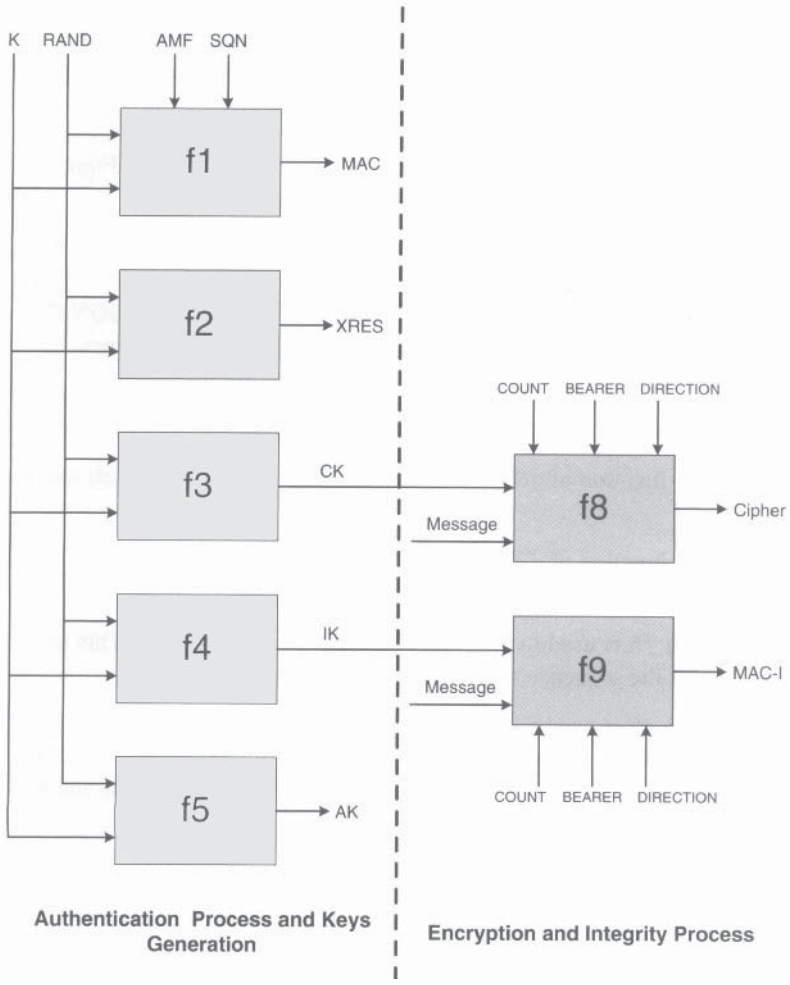


Figure 21.3 3G authentication and encryption mechanism.

authentication between the mobile phone and network operator, the inclusion of integrity check to ensure unmodified messages between mobile phone and operator, the use of longer encryption keys with some well known secure algorithms, the inclusion of several different parameters into the encryption algorithm to prevent attacks like replay attack and reflection attack, etc. Based on these changes, the security system of 3G is much more stronger than that of 2G systems. It is widely expected that this system can withstand most of the known attacks nowadays.

21.4 SHORT-RANGE WIRELESS SECURITY

21.4.1 Bluetooth Security

In this section, the authentication and encryption processes of Bluetooth will be discussed in detail. We also describe a recently discovered flaw in Bluetooth.

21.4.1.1 Bluetooth pairing process Figure 21.4 illustrates the processes of Bluetooth pairing [130]. Bluetooth pairing starts with the entry of PIN code. There are two kinds of devices. The first kind of devices has predefined (fixed) PIN code pre-installed. The other kind of devices allow entry of a variable size PIN code. The variable size PIN code ranges from 8 bits to 128 bits.

As illustrates in Figure 21.4, after the entry of PIN code at both side, the master generates a 128-bit random number (IN_RAND) and sends it to the slave. They will then make use of $E22$ algorithm to generate the $Kinit$ (initialization key) with the use of PIN code and slave BD_ADDR. At this point, the initialization key generation processes are regarded as completed. Further keys will be generated based on this initialization key.

As both of them possess K_{init}, they will then individually generate a 128-bit random number again (LK_RAND) and transmit this number to the other party after XOR'ing with K_{init}. With the knowledge of K_{init}, both sides' LK_RAND can then be derived. They will then be fed with opposite side's BD_ADDR into $E21$ algorithm to generate LK_Key. Together with self generated LK_RAND, self BD_ADDR and $E21$ algorithm, another set of LK_Key will be derived. The combination of both LK_Key will result in a combination link key K as shown in the middle part of Figure 21.4. The generation of link key K marks the end of link key generation processes.

With the common shared secret (link key K), this two entities can then authenticate each other to perform mutual authentication processes. First of all, one side will act as verifier and the other side will act as claimant. In the case shown in Figure 21.4, slave acts as verifier and master acts as claimant at the beginning. The slave will first generate and then transmit a 128-bit AU_RAND (challenge) to the master. The master will then perform calculation through $E1$ algorithm with the use of its own BD_ADDR, common link key K and the AU_RAND. The slave will carry out the same calculations. The results will consist of a 96-bit ACO (Authentication Ciphering

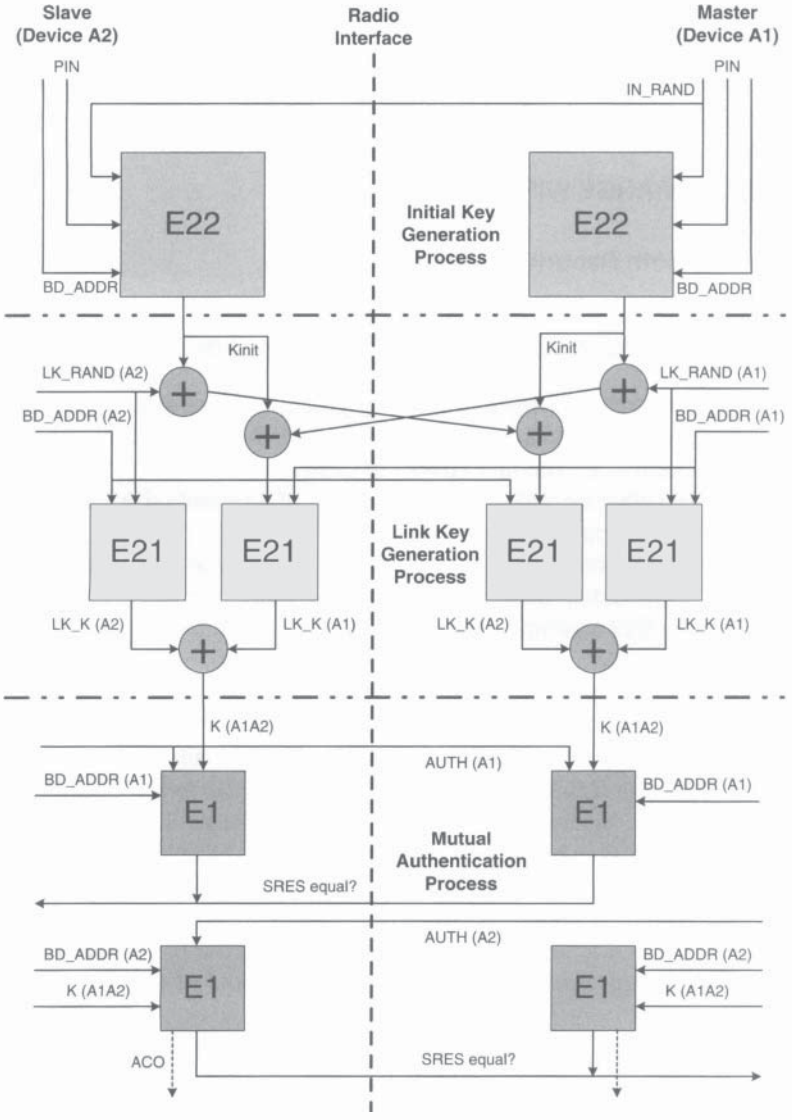


Figure 21.4 Bluetooth pairing process.

Offset) and SRES (Signed Response). The slave authenticates the master if their calculated SRES values match. Afterwards, the roles of verifier and claimant are reversed with the same authentication processes. The ACO of the last authentication process will be kept for the generation of ciphering later.

Up to now the whole authentication process is regarded as completed. The devices can then carry out communication if encryption is unnecessary. Both devices are regarded as paired or bonded at this moment.

21.4.1.2 Encryption Processes The data encryption process will be discussed in detail in this section. There are two kinds of messages to be considered. The first one is the unicast message that involve two communicating parties—a master and a slave. The second one is the broadcast message that involves a master and multiple slaves. Generally speaking, with the use of different encryption keys for different master-slave pairs, it is impossible to carry data broadcasting. Thus, a single shared link key has to be generated for the piconet for broadcast messages. This involves the generation of master key (called K_{master}).

The generation of K_{master} involves the use of $E22$ algorithm together with use of the two locally generated LK_RAND. This key will then be sent to each slave individually by XOR'ing it with the result of another generated RAND (send to the slave in plaintext) and link key K in algorithm $E21$. For individual slave which has the knowledge of RAND (received from master in plaintext) and the individually shared link key K with master, it can calculate the result and get back the K_{master}.

Now, let us consider the encryption processes [130]. Figure 21.5 illustrates the idea of Bluetooth encryption processes. As can be seen, the link key K, a random number and COF (Ciphering Offset) will be processed by algorithm $E3$ to generate the cipher key Kc. COF will be the exactly the same as the last authenticated ACO in case of encryption between two communicating parties. However, in case of broadcasting, COF will be derived from the master BD_ADDR. Similarly, the link key K here refers to link key between two communicating parties in case of encryption between them. On the other hand, K refers to master key in case of broadcasting.

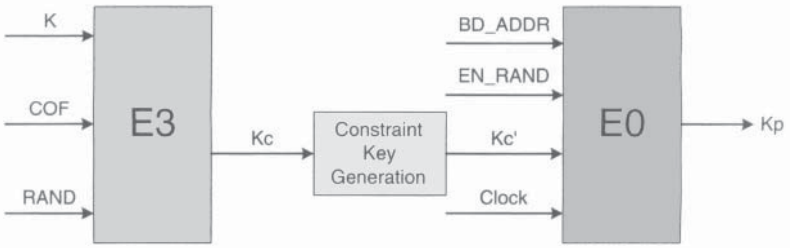


Figure 21.5 Bluetooth encryption processes.

The generated Kc will be 128-bits in size. However, due to some regulations (e.g., export control), sometimes it has to be restricted to be smaller than 128-bits. In this

case, it has to pass through constraint key generation algorithm to produce a smaller size constraint cipher key Kc'. This key will then be fed into algorithm $E0$ together with the BD_ADDR, EN_RAND and the clock to produce the payload key Kp. This key will then be the actual key used to encrypt the message to produce cipher at the sender side and decrypt the cipher to produce message at the receiver side.

We can see that the authentication and encryption processes of Bluetooth are quite involved. The keys are generated from the initial PIN code entry, which is then used to construct the initialization key, link key, cipher key, constraint cipher key and finally payload key used in encryption.

21.4.1.3 Discovered Vulnerability It is remarkable that Bluetooth security system was recently found to be flawed [317]. Specifically, the flaw is in the pairing process. The attackers are possible to make use of brute force attack to crack the PIN code if it is possible for them to eavesdrop the whole pairing process. First of all, let us review the messages transmitted over the air during the pairing processes. (The readers are referred to Figure 21.4 also.)

1. The master device sends a 128-bit IN_RAND to slave in plaintext.
2. The master device sends a 128-bit XORed K_{init} and LK_RAND(A1) to slave.
3. The slave device carries out the same actions. That is, it transmits 128-bit XORed K_{init} and LK_RAND(A2) to master.
4. The slave sends a 128-bit AU_RAND (challenge) to master in plaintext.
5. The master responses with a 32-bit SRES in plaintext.
6. The master sends a 128-bit AU_RAND (challenge) to slave in plaintext.
7. The slave responses with a 32-bit SRES in plaintext.

There are in total seven messages expected in the pairing processes. It is assumed that the attackers captured all these messages and know the source and destination of these messages. At this point, the attackers can then, in a brute force manner, try all the combinations of PIN code. With the knowledge of IN_RAND and BD_ADDR, the K_{init} can then be calculated through $E22$ algorithm. The K_{init} can then be used to perform XOR operation to derive two LK_RAND values. They will then be fed into $E21$ algorithm together with their BD_ADDR to derive suspected link key K after combining two LK_Key. Finally AU_RAND and SRES can be used to verify the correctness of the guessed value. If SRES can be derived correctly, the guessed PIN code is correct. With the use of some optimization of the SAFER+ algorithm which is the basis of most of the algorithm used in Bluetooth, 4-digit PIN can be resolved in less than a second and the 7-digit PIN can be resolved in less than an hour.

Some people may argue that this attack only affects the pairing processes. As long as this process is kept secret, this attack can be avoided. Unfortunately, the attackers

are possible to spoof the BD_ADDR of one of the paired user and request for the re-pairing processes. This is possible as Bluetooth's design allows either one side of the paired devices to lose the link keys due to whatever reasons. As a result, once the users re-do the pairing processes, there is an opportunity for the attackers to carry out the above procedures.

21.4.2 IEEE 802.11b Security

21.4.2.1 Overview IEEE 802.11b is widely used nowadays. Early versions of IEEE 802.11 has mandated the use of WEP (Wired Equivalent Privacy) for authentication and encryption of the messages. However, it has been proven flawed and the keys involved can be re-generated relatively easily. IEEE 802.11i is then being developed. In the mean time, some of the components including TKIP and IEEE 802.1x security framework that will be adopted in IEEE 802.11i are used until the final version is released.

21.4.2.2 WEP and its vulnerabilities Let us first review the mechanisms used in WEP. Afterwards, its vulnerabilities will be elaborated.

Figure 21.6 illustrates the key idea of WEP [27]. As can be seen, the shared secret key will be combined with the initialization vector (IV) to form a block which will then pass through the widely used stream encryption algorithm RC4 to form a keystream. At the same time, the CRC of the plaintext data will be computed and combined with the plaintext data to form a block. The two blocks that have been mentioned will then XOR'ed together to form the ciphertext. After combining with the original IV used in the whole process, the whole block of ciphertext can be sent to the receiver side.

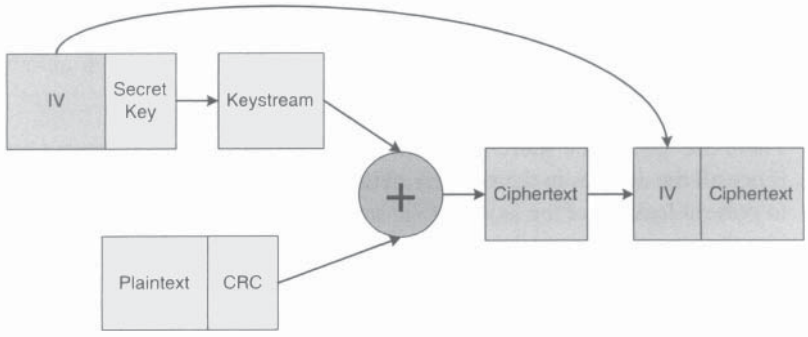


Figure 21.6 WEP mechanism.

At the receiver side, the message can be decrypted by first getting the IV from the received data block and combined it with the secret key to get the same keystream

block as that of sender side. This keystream can then be used to XOR with that of cipher to get back the original plaintext message with CRC.

This simple processes form the basis of WEP in encrypting and decrypting the plaintext message block in original IEEE 802.11b design. Parties in communication are required to know the shared secret key beforehand in order to encrypt and decrypt the message. However, this approach, in fact, contains some known flaws and problems. The major ones are listed below.

- WEP involves the use of 24-bit IV which will then be combined with 40-bit or 104-bit key to form a block to feed in RC4 algorithm to form a keystream. However, since the size of IV is small, the keystream will be repeated within hours in heavy traffics environment. It will then be subjected to keystream attack and replay attack. Keystream attack involves the XOR'ing of the two chipertexts using the same keystream. As a result, XOR'ing of two data blocks will be resulted. If one of the data block is known, the other data block can also be recovered. This is often possible due to the existence of some known control message data exchange at upper layers. On the other hand, replay attack is possible because IV out of order usage is possible due to its limited size. Consequently, if certain plaintext and cipertext pair is known, the corresponding keystream can also be recovered which can then be used to create new packet without the knowledge of WEP key.

- Forgery attack is also possible on WEP by simple flipping bits of the packet. It is because WEP ICV (Integrity Check Value) makes use of CRC as a mechanism for integrity check of plaintext. Thus, simply flipping the bits of packets allows the decrypted plaintext to be changed with a valid CRC value.

- FMS attack targets on certain weak keys of WEP no matter it is 40-bit or 104-bit key. In this method, large amount of encrypted packets are captured. During the encryption of these packets, certain concatenation of IV to the key will create weak keys. Consequently, by capturing large enough amount of encrypted packets, the key can then be recovered easily.

- There are also some more problems regarding the key management issues. Since all the devices in the wireless network shares the same key, it is difficult to prevent leakage of the keys. At the same time, it does not have a measure to identify a particular wireless user in the network as they all use the same key with the same authentication method.

21.4.2.3 TKIP (Temporal Key Integrity Protocol) In the previous section, some of the known flaws and known problems in WEP currently used in IEEE 802.11 have been discussed. In view of these problems, IEEE 802.11i task group was developed to find a replacement for WEP[2]. However, in the meantime, some of the

[2]Originally, this was known as WEP2. However, it is renamed as WPA (Wi-Fi Protected Access).

methods have to be adopted to tackle the serious security issues suggested previously, e.g., some of the attacks that will lead to the leakage of the key. Thus, WPA comprises both TKIP and 802.1x which aims at tackling the encryption key issues and access control issues, respectively [27].

First of all, let us start with TKIP, which mainly targets at solving the problems of the encryption key problems in WEP. It is just an interim solution which aims at providing a patch up solution on existing hardware with simple software and firmware upgrade in order to provide backward compatibility. TKIP was designed as shown in Figure 21.7.

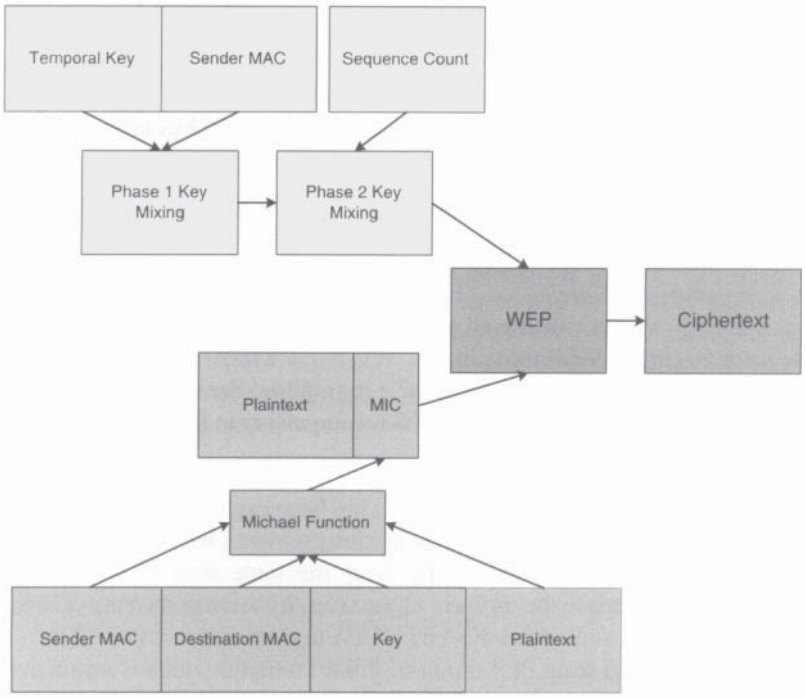


Figure 21.7 TKIP mechanism.

Figure 21.7 illustrates the idea behind TKIP. There are two keys involved. They are 128-bit encryption key (Temporal Key) and 64-bit MIC (Message Integrity Code) key. Their functions are as follows. The temporal key will first be combined with the sender's MAC address to form a intermediate key in phrase 1 which will then be fed into phrase 2 to further combine with the sequence number. Since the sequence number, in fact, depends on the packets transmitted, a 128-bit per-packet key will be generated after phrase 1 and phrase 2 key mixing process. This 128-bit will be broken into 24-bit IV and 104-bit WEP key. Finally, original WEP procedures are adopted as usual. On the other hand, the plaintext will be mixed with several parameters

including 64-bit MIC key, Sender and Receiver MAC addresses to form a plaintext and MIC block as shown in Figure 21.7. This block is then fed into the original WEP procedures shown in Figure 21.6.

The above procedures can effectively solve the problems of WEP as supported by the following arguments.

- The packet integrity check no longer makes use of CRC. Instead, a one way hash known as Michael is used. This function makes use a number of items to form the hash value and this effectively solves the integrity attacks.
- FMS attacks, which take advantage on the weak key, are no longer feasible as the per-packet key does not correlate to the IV any more. In other words, the 104-bit key fields will not be the same even for the same 24-bit IV value is used. At the same time, the per-packet key will be different for different wireless users. It is because the two-phase key mixing has taken into account the value of MAC address and sequence number.
- Other than the attacks stated above, with the use of sequence number, replay attack can be avoided. With the avoidance of the correlation between the per-packet key and IV, key stream attacks are much more difficult to be achieved.

It is currently widely considered that TKIP has effectively solved some of them problems in originally WEP idea. In long run, a much more secure encryption idea should be adopted. As a result, IEEE 802.11i proposed the use of AES (Advanced Encryption Standards) for encryption. AES is compulsory in IEEE 802.11i standards, while TKIP is just optional.

21.4.2.4 802.1X access control As it has been explained in previous section, WEP not only contains problems in its encryption problems, but also uses very simple mechanism in authentication process by using the same shared key in the whole network. This also leads to the problem of user identification in the network. In view of this, 802.1x access control is adopted in WPA to tackle these issues [27].

802.1x was adopted from PPP (Point-to-Point Protocol) which is originally used in Modem dial up and later extended to ADSL. It provides point to point connection between two entities with the username and password as authentication measures. In order to provide more authentication options on top of PPP, EAP (Extensible Authentication Protocol) was proposed.

802.1x uses EAP to provide authentication and identification in both wired and wireless environments. The three main components involved and their corresponding functions are summarized below.

- Supplicant is the client who attempts to access the network.
- Authenticator is the one trusted by the network and acts on behalf of the authentication server to authenticate the supplicants. As a result, it can be regarded as a *man*-in-the-middle and this role is usually played by the access point.

- Authentication Server is the server which contains all the authentication information of the network and makes decision on whether successful authentication occurs. This is usually performed by a RADIUS server.

Figure 21.8 illustrates the idea of 802.1x. First of all, the supplicant sends an EAP start message to authenticator to show its intention to access the network. The authenticator will first put the supplicant in controlled ports in unauthorized state. It sends the EAP Request/Identity message to supplicant. The supplicant then supplies its identity through EAP Request/Identity message back to authenticator which will then forwards the message to authentication server. The communication between the authenticator and authentication server is through an uncontrolled port since they are both trusted in the network. After the EAP-Request(challenge) and the subsequent challenge/response as shown in the figure, the authentication server determines whether the supplicant is authorized to access the network. The authentication server then sends this decision to the authenticator. The authenticator will then put the traffics of supplicant into controlled port authorized state if the authentication is successful.

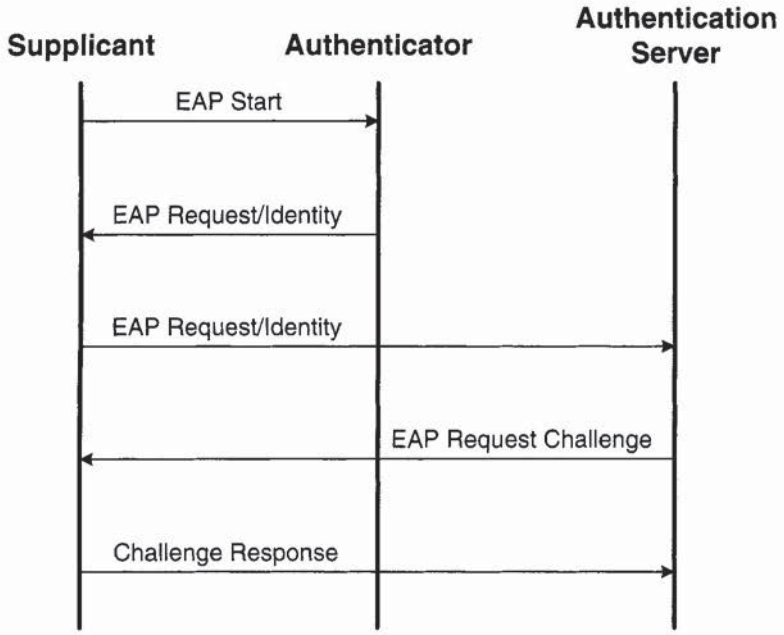


Figure 21.8 802.1x mechanism.

From the processes that have just been discussed, the authentication server authenticates the supplicants through a number of procedures. By doing so, different keys can be used by different supplicants with appropriate EAP algorithms that suit

the security needs of the network. At the same time, the identity of the users can be recorded for easily billing later. On the other hand, with the use of 802.1x, periodic rekeying is also possible. Generally speaking, this idea effectively solves the problems that have been suggested.

21.5 PRACTICAL ILLUSTRATIONS—WIRELESS LAN SECURITY AUDITING

In previous sections, different technologies security issues have been discussed. Most of these measures have already been adopted into the products in the market. In this section, a distributed wireless security auditor (DWSA) which has been implemented to detect the rogue access point inside will be discussed to illustrate a practical way to detect security threat by using WLAN [41].

There are a number of attacks in WLAN related to the access point configuration and management. In a corporate environment, even though access point configuration and management processes are usually carefully implemented, they are still susceptible to attacks like insider attack in which rogue access point may be setup by some authorized users whose original intention may just want to provide convenient to other users with wireless access capability. In view of this, DWSA has been implemented to detect the rogue access point. Figure 21.9 illustrates the idea behind DWSA. We can see that the clients become distributed wireless sensors and monitor the existence of any access point located around them. Their functions are summarized below.

- Gather information about the access point searchable around wireless sensor.
- Get wireless client location information by using triangulation which makes use of the received signal strength or using GPS system integrated inside the wireless client
- Periodically report the information gathered to wireless security auditor server through SSL connection.

Moreover, the wireless security auditor server contains much sensitive information including the list of authorized access points and their corresponding physical location inside the organization, the security policy of the organization and the building layout of the organization. This server is used to audit the information gathered from the distributed wireless clients. The main functions of the server are summarized below.

- Authenticate distributed wireless clients that provides information to the server through SSL connection.
- Determine whether rogue access point exists and its corresponding location based on the information provided by various distributed wireless sensors and

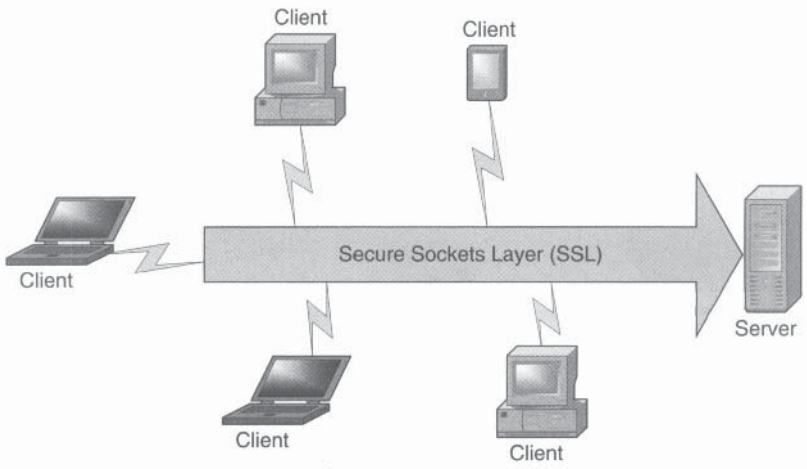


Figure 21.9 DWSA system.

the information originally stored inside the database of the system, e.g., the organization security policy and location of authorized access point.

DWSA architecture relies on the distributed wireless sensors (wireless clients) to gather information about the security inside the corporate network and then send back the information to the processing server through SSL. This allows secure information transfer back to the server. At the same time, only little processing power is required on the clients since all the works are done by the server. The main advantage of this architecture is that it not only detects whether there exist rogue access point, it also tries locating it by first locating the wireless clients itself inside the building.

In general, there are two methods of locating the rogue access point in the network. The first method is by GPS integrated inside the clients. However, this is sometime impossible as not all the clients are integrated with GPS. The second method includes the use of triangulation method. Three-dimensional triangulation environment, just like the case in corporate network with several floors, can be done whenever four fixed access points location and their corresponding distance to the client are known beforehand. The idea behind is similar to find the intersection point of four sphere with the known radii and their center positions. The distance between a fixed access point and a client can be estimated through the received signal strength. With the idea of the current location of wireless client, any other rogue access point location can be estimated by the client through the received signal strength to that access point. Figure 21.10 illustrates an example scenario of how DWSA locates the rogue access point in the network.

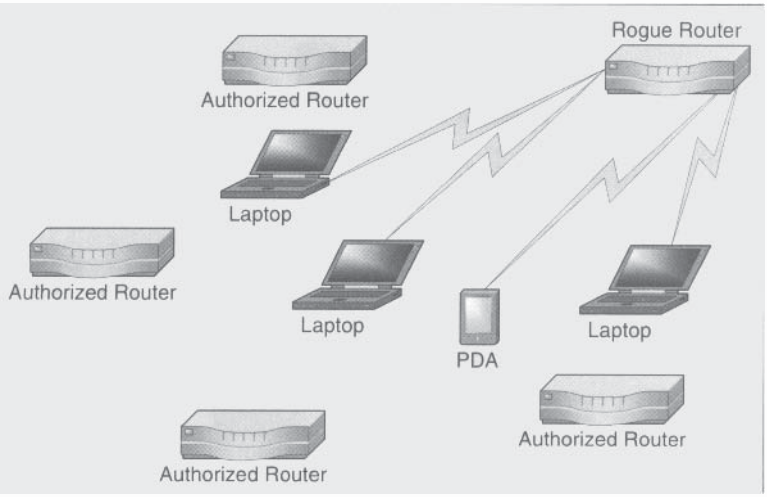


Figure 21.10 DWSA example scenario.

21.6 SUMMARY

In this chapter, different wireless technologies' methods on handling security issues have been introduced. First of all, we discuss different considerations with the main aim at providing authentication, privacy, etc., on wireless medium which is open and public in nature.

Afterwards, specific security issues on cellar network have been suggested with the unpublished security algorithms used in second generation cellular network, e.g., GSM network and CDMA network. However, the unpublished security algorithms are found to be flawed. In view of these, third generation cellular network security has been designed with adoption of the solutions to the known weakness in second generation cellular network security from some published and proven secure algorithms.

After discussing the cellular security issues, short range wireless security issues are described. Short range wireless networks involve quite different characteristics compared with that of cellular network in terms of range, capacity, client portable devices configurations, etc. As a result, they also have different considerations in their security designs. We describe IEEE 802.11 WLAN's and Bluetooth WPAN's methods in tackling security issues. Finally, this chapter is concluded with a practical illustration on a proposed WLAN architecture which is possible to audit fake access point with their corresponding location.

In this chapter, the authors have walked through the security issues in different kinds of cellular and wireless networks. One point in common is that the open

and public nature of wireless medium has caused many security problems. Many algorithms have been suggested and adopted to tackle these problems. However, it is generally known that the solutions adopted are hardly perfect due to reasons like implementation difficulties, specific countries export regulation, etc. As a result, there are constantly some flaws found and new types of attack thought of by the attackers. It is obvious that security is an ever on-going battle.

PROBLEMS

21.1 Briefly elaborate the common problems and considerations in wireless communication network security design.

21.2 Briefly explain the security measures adopted in GSM networks and the counter-measures adopted by the attackers in breaking such systems.

21.3 Briefly explain the reasons why WEP is flawed in nature and how WPA has improved it.

21.4 Briefly discuss the security measures adopted by Bluetooth and discuss how it is broken recently.

This Page Intentionally Left Blank

PART VI

MOBILE COMPUTING APPLICATIONS

This Page Intentionally Left Blank

CHAPTER 22

VOIP ON WIRELESS

There is no index of character so sure as the voice.

—Benjamin Disraeli:Tancred, II, 1847

22.1 INTRODUCTION

Voice communication is still the dominant form of data transmission in wireless networks. As VoIP over the Internet has proliferated, people naturally expect the same kind of service to be available in wireless environments.

Interoperability:

Indeed, VoIP (Voice over IP) over wireless has become a hot research topic recently. The reason is that it will be much more cost effective to use an IP packet switching network to support voice traffic than a circuit switching network. Specifically, it is not required to have a fixed reservation of resources through the whole path from the origin to the destination. At the same time, some more advanced encoding algorithms can be used to support the voice traffic to further lower the bandwidth required.

Wireless Internet and Mobile Computing. By Yu-Kwong Ricky Kwok and Vincent K. N. Lau
Copyright © 2007 John Wiley & Sons, Inc.

However, there are many problems required to be solved in using wireless VoIP due to interoperability issues. The mainstream wireless and wired voice traffics nowadays are supported by circuit switching network like GSM, CDMA, third generation network and PSTN, in which operators have invested lots of money. As a result, in order to provide real-time voice conversation service to users, wireless VoIP is required to provide interoperability to the circuit switching network, including signal conversion, voice encoding conversion, etc. This is usually done with the addition of a gateway into the infrastructure of the backbone network. Sections 22.2 to 22.4 illustrate the idea of cooperating GSM with H.323 IP network to provides users with a choice of using either technology for telecommunication services.

On the other hand, for the case of using VoIP over WLAN, it may not be a good idea to use the original protocol stack of IEEE 802.11 and/or use it together with other data traffics. However, due to interoperability issues, a large change to IEEE 802.11 especially for VoIP traffics is impossible. As a result, some other methods have to be adopted to mitigate the interoperability issues, while at the same time improve the performance. This is further discussed in Section 22.5.

Performance:

In iGSM, which is a system involving the cooperation of GSM and H.323 VoIP networks, the access delay for call delivery and the number of the occurrence of misrouting due to user mobility are good metrics to evaluate the performance of iGSM, as these are the two problems required to be solved with the use of iGSM. We discuss these two problems in Section 22.5.

On the other hand, using VoIP over WLAN, the number of VoIP streams that can be supported, packet access delay and packet loss are three main metrics for evaluating the performance of VoIP over WLAN. Section 22.5 illustrates some practical ideas on improving the performance related to these three metrics.

22.2 IGSM VOIP APPROACH

iGSM is a kind of value-added service on GSM, allowing users to choose either GSM network or H.323 IP network with the use of handsets or H.323 terminals (e.g., PC), respectively, to access telecommunication services [295]. In other words, users with iGSM accounts are able to make/receive calls using their own handsets, or make/receive calls with the same mobile station ISDN (MSISDN) without their handsets by using H.323 terminals within the IP network. As a result, user mobility can be supported.

Similar to iGSM, there are also some other approaches which propose the integration of GSM network and IP network, such as TIPHON [107] and GSM on the Net [136]. The main difference between these approaches and iGSM is that iGSM targets on providing value-added services to GSM users to support user mobility with the cooperation of H.323 VoIP network, in which no specific wireless capabilities are provided to GSM handsets. However, BSC/BTCs in IP networks and GSM/BTSs in

corporate offices are proposed to provide wireless access capabilities to GSM handsets to use VoIP services provided within the IP networks in TIPHON and GSM on the Net, respectively.

22.3 IGSM SYSTEM ARCHITECTURE

The architecture of iGSM is illustrated in Figure 22.1 [295]. In the figure, it can be seen that there are two main parts of the backbone network—GSM network and H.323 IP network. Actually, the GSM network portion is unmodified. However, an iGSM gateway is added into the system to act as an interface between the GSM network and H.323 IP network. This gateway is regarded as a member of the IP network and performs several functions similar to that of a traditional H.323 gateway. These functions are summarized below:

- GSM/PSTN/IP Call Control functions including call-setup and call-release.
- GSM MAP protocol and H.225 RAS (Registration, Admission and Status) protocol translation between the GSM network and H.323 IP network.
- Provision of media translation to cope with QoS differences between the IP network and circuit switching network.

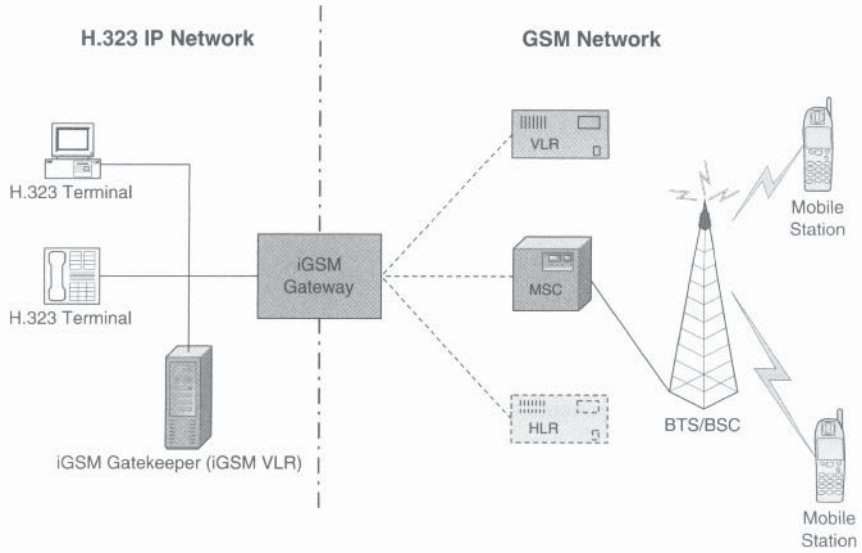


Figure 22.1 iGSM architecture [295].

On the other hand, an iGSM gatekeeper is also present inside the H.323 IP network. In the traditional H.323 network, which is shown in Figure 22.2, the incorporation of

a gatekeeper is optional to implement functions like address translation, bandwidth allocation, admission control, zone control, etc. However, in the case of iGSM, the iGSM gatekeeper is implemented inside the H.323 IP network to function as a VLR for users visiting the IP network. A unique ISDN is assigned to each iGSM gatekeeper to let the HLR know that it is a VLR. Being the roles of a gatekeeper and a VLR, the iGSM gatekeeper maintains an iGSM subscriber list and handles appropriate address translation. In other words, it performs functions of the original H.323 gatekeeper and also the functions of the VLR in the iGSM network. We discuss the detailed procedures of registration, deregistration and call-setup in subsequent section.

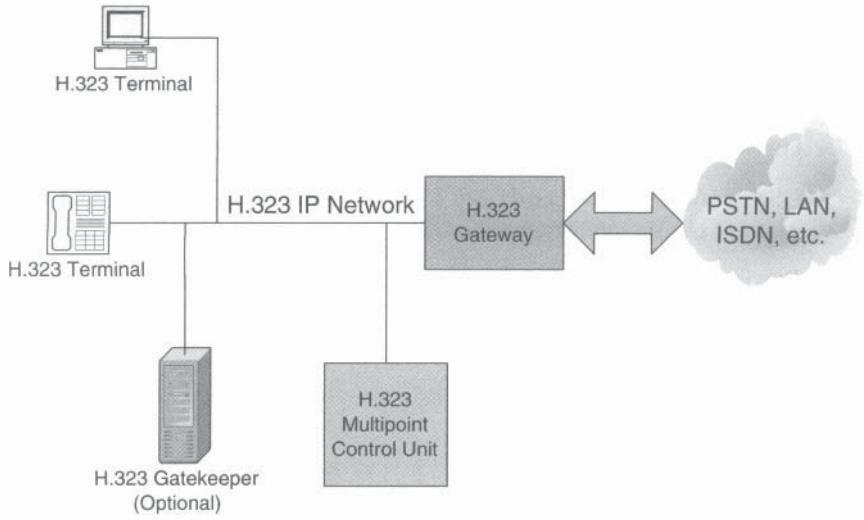


Figure 22.2 H.323 architecture.

22.4 IGSM CALL PROCESSING

In previous sections, the idea of iGSM and its architecture have been discussed. With the additional components, namely the iGSM gateway and iGSM gatekeeper, cooperation services can be provided to iGSM users to use either GSM service or H.323 service. In this section, the detailed procedures involved in the registration of GSM users to the H.323 IP network, explicit deregistration and call setup and delivery are discussed.

22.4.1 Registration Procedures

When iGSM users are within the GSM network, the registration procedures follow exactly the procedures in the traditional GSM network using the GSM MAP protocol.

However, if the users move to the H.323 IP network, the registration procedures are different. Figure 22.3 illustrates the registration procedures required for an iGSM user in the IP network. The details of the procedures are described below:

1. The whole procedures start by the user's moving from the GSM network to the IP network. Within the IP network, the user switches on the H.323 terminal and inputs the MSISDN together with the password. These actions activate the iGSM VoIP service. The H.323 terminal then sends an RRQ (RAS Registration Request) message, which includes the user's MSISDN and password, to the iGSM gatekeeper.

2. On receipt of the MSISDN and password of the user, the iGSM gatekeeper acts as a VLR to verify the password and authenticate the user. After authentication, the gatekeeper then sends an IRQ (Information Request) message to the iGSM gateway. This message includes the information about the user, including operation type (GSM UpdateLoc), VLR's ISDN assigned by the HLR, IMSI of the user, and the MSC number (i.e., the corresponding H.323 terminal address). This information is then translated into standard GSM MAP protocol UPDATE_LOCATION and is sent to the HLR.

3. IMSI is an internationally unique number for a subscriber. With this number, the HLR is able to get the user profile of the iGSM user. This profile is sent back to the iGSM gateway using the standard GSM MAP message INSERT_SUBS_DATA, which is then forwarded to the iGSM gatekeeper using an IRR (Information Request Response) message. Afterwards, the iGSM gatekeeper stores the user profile in its database. In response to the user profile message, the iGSM gatekeeper sends an acknowledgement message using IRQ to the iGSM gateway, which then forwards the acknowledgement message to the HLR.

4. At the same time of forwarding the user profile to the new VLR (i.e., the iGSM gatekeeper), the HLR also updates the old VLR using some deregistration procedures. These include the transmission of CANCEL_LOCATION to the old VLR with the IMSI of the user. The old VLR then responds with an acknowledgement of the CANCEL_LOCATION message to indicate the deregistration of the user from it.

5. At this point, the registration procedures are nearly completed. The HLR then sends an acknowledgement of the UPDATE_LOCATION message with its own HLR number to the iGSM gateway to indicate the success of the registration. This message is then forwarded to the iGSM gatekeeper using an IRR message. Finally, the gatekeeper sends an RCF (Registration Confirmation) message back to the H.323 terminal to indicate the finish of the registration.

After the above procedures, the iGSM user can now use an H.323 terminal device such as a PC to make/receive calls with the same MSISDN.

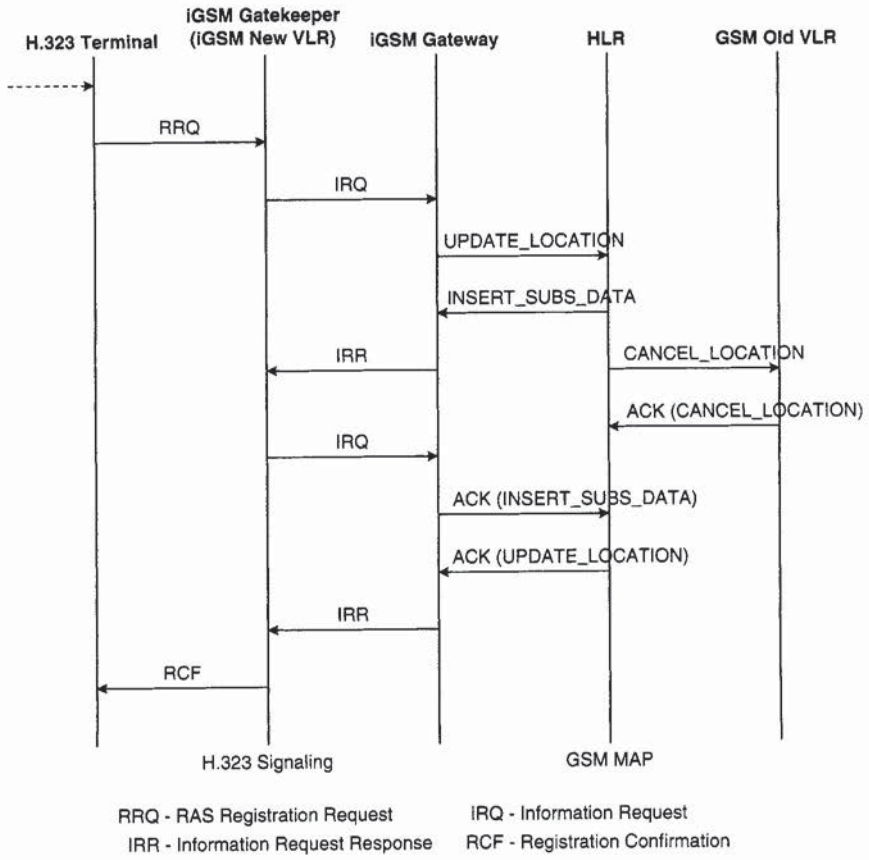


Figure 22.3 iGSM registration procedures.

22.4.2 Deregistration Procedures

As discussed in the previous section on iGSM registration procedures, the registration into the IP network is in fact the deregistration from the GSM network. For iGSM deregistration procedures, the situation is reversed. When the iGSM user moves back to the GSM network, for example, by turning on the handset again, this action will deregister the user from the IP network. Figure 22.4 illustrates the procedures required in the deregistration of the iGSM user from the IP network. The details of the procedures are described below:

1. When the iGSM user turns on the handset again, the registration procedures follow the GSM MAP protocol. The handset registers to the VLR, which then informs the HLR with a UPDATA_LOCATION message. Afterwards, the HLR starts the deregistration procedures. It first locates the iGSM gateway and sends it a CANCEL_LOCATION message with the IMSI of the iGSM user so as to indicate that the user has left the IP network. The iGSM gateway then sends an IRR message to the iGSM gatekeeper with IMSI and CanLoc as the operation type. After that, the iGSM gatekeeper sends a URQ (Unregister Request) to the H.323 terminal.

2. The H.323 terminal responds with a UCF (Unregister Confirmation) message as an acknowledgement to the completion of deregistration back to the iGSM gatekeeper. After knowing that the H.323 terminal has finished the deregistration procedures, the iGSM gatekeeper responds with an IACK (Information Request Acknowledgement) back to the iGSM gateway, which then generates an acknowledgement to the CANCEL_LOCTION message back to the HLR to indicate the completion of deregistration procedures in the IP network.

22.4.3 Call-Setup Procedures

The above procedures describe the registration and deregistration processes of an iGSM user in the H.323 IP network. In this section, we describe the call-setup procedures. If the user is within the GSM network, the call origination and call delivery follow the standard GSM procedures. However, if the user is within the IP network, the call origination and call delivery follow the H.323 procedures. Figure 22.5 illustrates the call-setup procedures while the iGSM user is within the IP network and the call originated from another terminal (e.g., a PSTN or GSM terminal) is delivered to the H.323 terminal of the iGSM user [295]. The whole process is described below:

1. The caller, who may be using another GSM mobile station, dials the MSISDN of the iGSM user. From the MSISDN, the GMSC (Gateway MSC) of the mobile station can be located. As a result, the originating switch sends a SS7 IAM (Initial Address Message) to the GMSC to initiate the call.

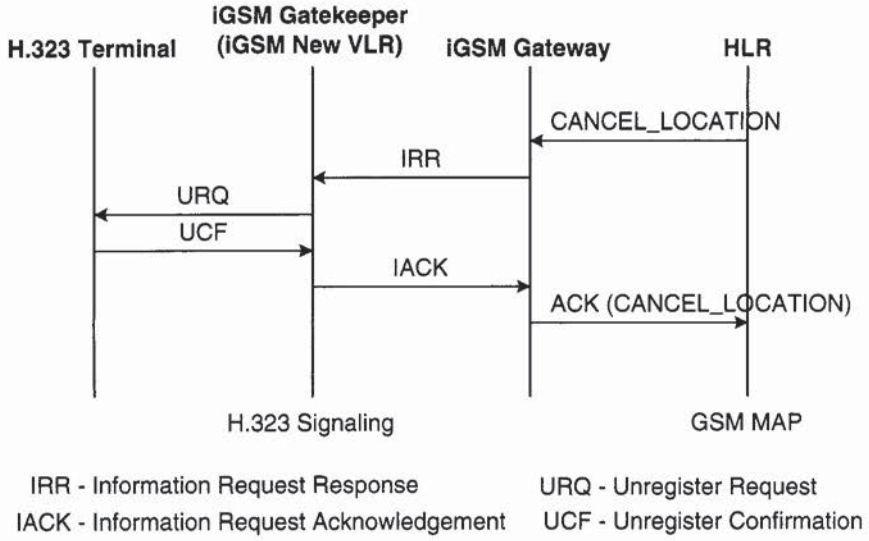


Figure 22.4 iGSM deregistration procedures.

2. On receiving the IAM message, the GMSC needs to find out the current location of the mobile station. Therefore, it needs to query the HLR to ask for the location by using the GSM MAP protocol—SEND_ROUTING_INFORMATION. This message also includes the iGSM user's MSISDN. With this MSISDN, the HLR is able to locate the record of the iGSM user and find out his IMSI. Afterwards, the iGSM user is found to have registered to the HLR through the IP network previously. Then the corresponding iGSM gateway and iGSM gatekeeper can be located. As a result, the PROVIDE_ROAMING_NO message is sent to the iGSM gateway, which is then translated to an RAS LRQ (Location Request) message with RoamNo as the operation type and IMSI of the iGSM user.

3. On receipt of the LRQ message, the iGSM gatekeeper, which is also the current VLR of the H.323 terminal, is able to locate the H.323 terminal (may be a PC) used by the iGSM user. As a result, the MSRN (Mobile Station Routing Number), which is derived from the iGSM gatekeeper and the H.323 terminal address and can be used to uniquely locate the H.323 terminal, is transmitted back together with an acknowledgement in an RAS LCF (Location Confirmation) message to the iGSM gateway. The gateway then sends back the MSRN as a payload in the acknowledgement of the SEND_ROUTING_INFORMATION message to the GMSC, which initiates the query of location of the mobile station at the beginning.

4. The GMSC, which has the unique location information of the H.323 terminal, sends an IAM to the iGSM gateway to show the intention to initiate a call. The IAM message is then translated by the gateway into an ARQ (Admission Request), which is then sent to the iGSM gatekeeper. The gatekeeper responds with an ACF (Admission Confirmation) message to show the possibility of setting up a connection within the IP network. Recall that one of the main functions of the iGSM gatekeeper is bandwidth allocation.

5. The iGSM gateway then sends an H.225 setup message to the H.323 terminal. If the H.323 terminal notifies that there is a call for it and that it would like to accept it, it then replies with a call proceeding message to indicate that it has received enough routing information for the call. Afterwards, the H.323 terminal initiates an ARQ message to iGSM gatekeeper to ask for the permission of accepting the call and to request for bandwidth allocation. Similar to the previous step, the gatekeeper replies with an ACF message.

6. Up to this point, the call-setup is prepared with enough signalling. The H.323 terminal then starts ringing to alert the iGSM user that there is a call for him. At the same time, the H.323 terminal also generates an alerting message to the iGSM gateway, which is then translated into an SS7 ACM (Address Completion Message) message. The ACM message is sent to the GMSC and then to the originating switch.

7. Whenever the iGSM user picks up the call from the H.323 terminal, a connect message is transferred to the iGSM gateway, which is forwarded to the originating switch through the use of SS7 ANM (Answer Message) messages. Finally, conversation between the two communicating entities can be started.

22.4.4 Problems in iGSM

There are two main problems in the use of iGSM. They are the GSM tromboning effect and misrouting due to user mobility [295].

Tromboning effect occurs when the called mobile station and the dialer mobile station are within the same city, but the GMSC is located at other city or country (it is assumed that both GSM MSC and iGSM gateway can function as a GMSC). In this case, it is required to route signals to other city and then transfer back. This kind of calling is expensive.

If the iGSM user is assigned with a GMSC, then the call-setup procedures discussed before are followed by the tromboning effect when the GMSC is in other city or country. However, in the case that the iGSM user is assigned with an iGSM gateway which functions as a GMSC, the following call-delivery procedures are resulted:

1. The originating switch (may be a telephone from a PSTN network) dials the MSISDN of the iGSM user and queries the corresponding iGSM gateway.

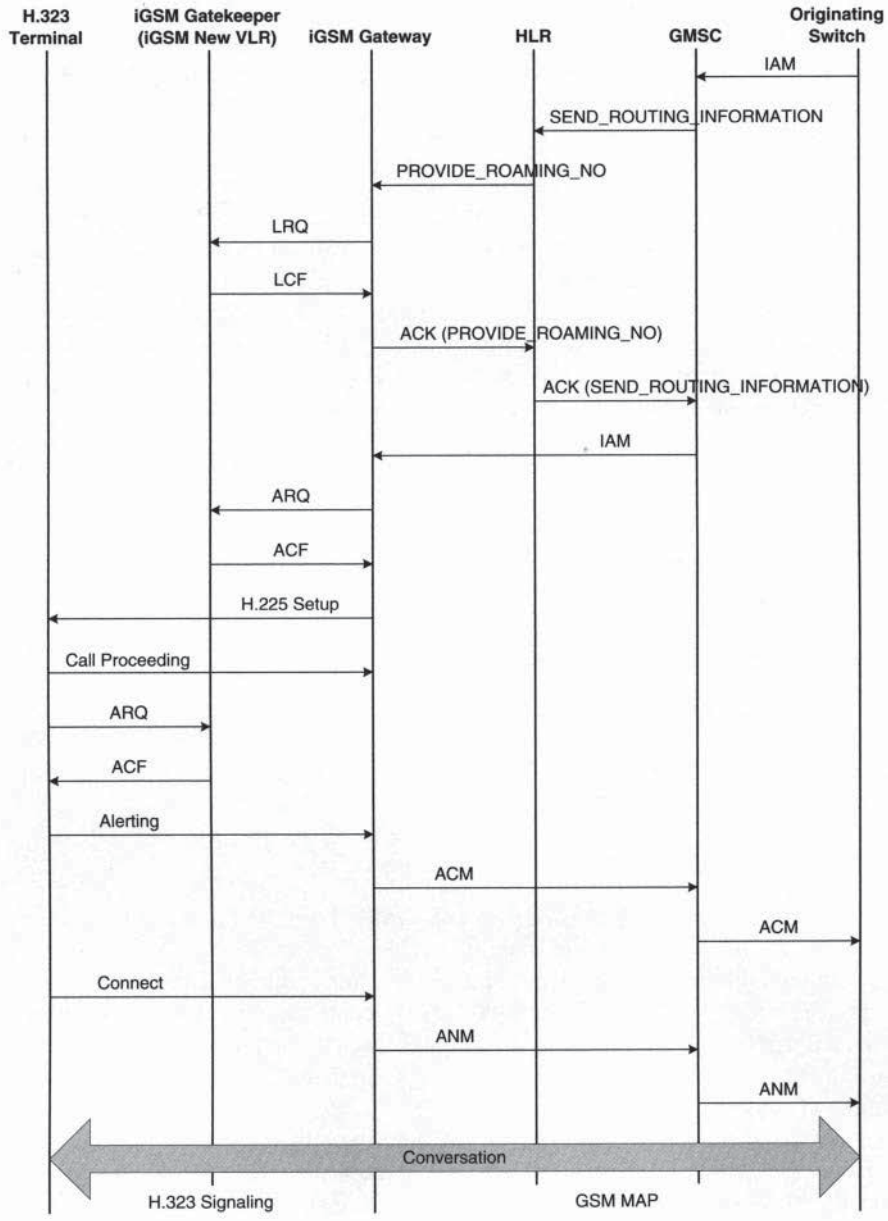


Figure 22.5 iGSM call-setup procedures.

2. The iGSM gateway then looks into its own iGSM gatekeeper (VLR) to see whether the iGSM user has moved from the GSM network to the IP network.

3. If the corresponding MSISDN cannot be found, then the query is forwarded to the HLR and normal GSM call-setup procedures are resulted.

4. If the corresponding MSISDN is found, then normal PSTN-IP call-setup is resulted with the use of normal H.323 call-setup procedures by the iGSM gateway.

Following the above procedures, the tromboning effect is effectively reduced due to the reason that the GSM network is not passed through at all with the iGSM gateway being used as the GMSC. This can result in a much lower call-setup cost than that of the original call-setup procedures. If both communicating entities are from IP networks, then the cost is much lower with the use of a pure VoIP environment. However, this solution may not be adopted by GSM operators if the iGSM gateway is not under their control.

After discussing the tromboning effect, we discuss the misrouting effect due to user mobility. Misrouting effect occurs when the iGSM user moves from the GSM network to the IP network with his mobile station turned on. After moving to the IP network, the registration procedures are carried on for the H.323 terminal which is located within the IP network. With this registration, the HLR remembers that user is in the IP network and the record in the original VLR is modified. After some time, the user leaves the IP network and moves back to the GSM network in the original location. Since his mobile station is still turned on, the mobile station does not explicitly ask for re-registration. In this case, the record in the HLR still indicates that the user is in the IP network, and thus misrouting occurs if someone makes call to the user.

Misrouting can be solved by two measures—implicit registration and explicit registration. Implicit registration occurs when the user dials a call to others. In this case, the VLR will figure out that the record of the user is not present and ask for re-registration. On the other hand, implicit registration also occurs when the user moves to other GSM network or IP network. In this case, registration is forced by the network.

However, the implicit registration scenarios described above actually occur by chance. Therefore, the user should be responsible for using explicit registration to inform the network. This involves turning off of the mobile station within the IP network and then turning it on again after getting back to the GSM network. If this is not possible, then the network can automatically turn off the mobile station if registration from the IP network is detected. With the use of these methods, the misrouting problem can be effectively solved.

22.5 PRACTICAL ILLUSTRATIONS—PROBLEMS AND SOLUTIONS FOR VOIP OVER IEEE 802.11

In previous sections, the integration of GSM network and H.323 VoIP network—iGSM, has been discussed with the detailed procedures of registration, call-setup, deregistration, and the associated problems and solutions of introducing VoIP into the GSM network. In this section, we discuss in detail a practical scenario on the use of VoIP over WLAN. We also discuss how to improve the performance of carrying VoIP traffic in a WLAN environment, and the associated problem when there is also other data traffic [345].

First of all, let us reiterate some of the ideas of WLAN. In IEEE 802.11 WLAN, there are several standards introduced, including a, b, g, etc. IEEE 802.11a/g allows a throughput from 6 Mbps to 54 Mbps, while IEEE 802.11b allows a throughput from 1 Mbps to 11 Mbps. Using GSM 6.10 as the codec, it requires a capacity of 13.2 Kbps, and a frame is generated every 20 ms. As a result, with the use of IEEE 802.11b WLAN and GSM 6.10 codec, it should be able to support around 800 of this kind of VoIP stream. However, it is found that the number of VoIP streams supported is much less than this value (around 12 streams can be actually supported).

There are several reasons behind the discrepancy of the two numbers. One of the main reasons is due to the overhead introduced in packet headers. For example, the use of RTP/UDP/IP introduces 40 bytes of packet overhead, while a packet usually involves 10-30 bytes of payload for VoIP applications. This introduces more than 50% of overhead. At the same time, various MAC control mechanisms (e.g., backkoff time, various IFS, etc.) and the PHY layer overhead (e.g., permeable, etc.) can introduce an overhead of more than 800 ms. As a result, the efficiency of VoIP over WLAN is quite low (generally not more than 5%).

In view of the above problems, the voice M-M (Multiplex-Multicast) scheme is proposed [345]. Figure 22.6 illustrates the original idea of VoIP over WLAN, and Figure 22.7 illustrates the modifications made in the M-M scheme. In both cases, a voice gateway is located at the end of each side of the Internet. This gateway is a requirement in the H.323 standard for address translation, call admission control, call signal routing, etc. However, the multiplexer is assumed to be present only in the voice gateway in the M-M scheme.

Originally, without the use of the M-M scheme, VoIP packets are routed from one source to one destination in a unicast manner. However, with the use of the scheme, the packets originated from multiple sources (from S1 to Sn in Figure 22.7) are generated individually and passed to the voice gateway through the access point (AP). In the gateway, multiple packets targeted for the same Basic Service Set (BSS) are combined and multiplexed into one packet with the destination address changed to the multicast address of the BSS. The new packet is then routed through the Internet and arrives at the access point of the designated BSS through another voice gateway. The access point then multicasts the packet within the BSS. In this case, the destination

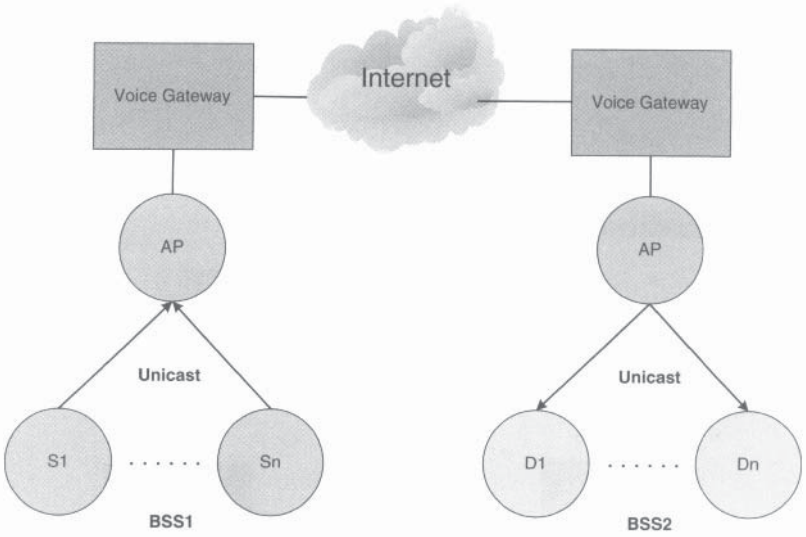


Figure 22.6 VoIP over WLAN.

host receiving the packet looks into the payload to determine whether there is VoIP packet designated for it. The whole process is shown in Figure 22.7

With the use of the M-M scheme, the number of streams supported within one BSS is reduced from $2n$ (n uplink unicast streams and n downlink streams) to $n + 1$ (n uplink unicast streams and 1 downlink stream). As discussed earlier, one of the major reasons for the poor performance of VoIP over WLAN is due to the significant header overhead introduced in various layers. However, the M-M scheme can effectively reduce the headers within each BSS by nearly 50%. As a result, performance improvement of around 90% to 100% is recorded through detailed analysis and simulations [345]. However, one of the limitations of the scheme is that the original security measure used for the unicast stream may not be applicable to the scheme. At the same time, multicast in IEEE 802.11 does not support ARQ. As a result, the performance of the scheme will be varied according to the collision rate.

After discussing the details of the M-M scheme in improving the performance of VoIP over WLAN, we discuss the effect of the presence of TCP data traffic, which is ignored in previous discussion. In fact, TCP traffic can greatly affect the performance of VoIP over WLAN. Figure 22.8 illustrates the scenario that we are going to discus. In the figure, there is an access point in which there are $2n$ VoIP streams and one FTP traffic with uplink and downlink streams. The TCP traffic can affect the VoIP traffic in two ways—the queue in the AP and medium access contention between nodes [345].

As for the queue in the AP, the general implementation of commercial access points involves the deployment of FIFO queue which is shared by all kinds of traffics.

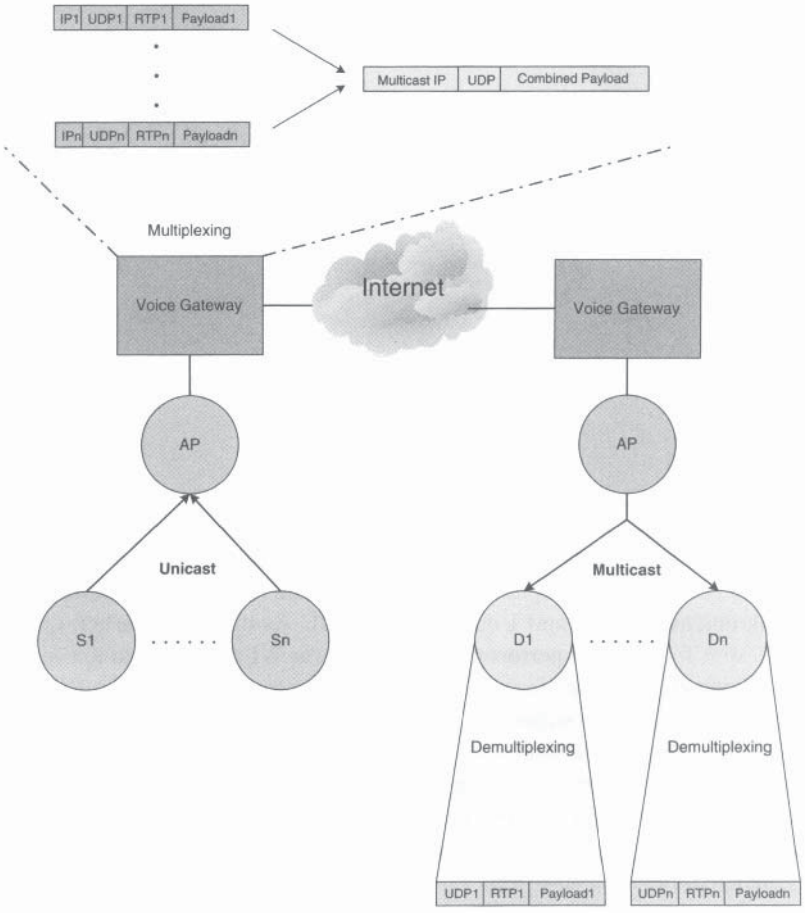


Figure 22.7 VoIP over WLAN M-M scheme.

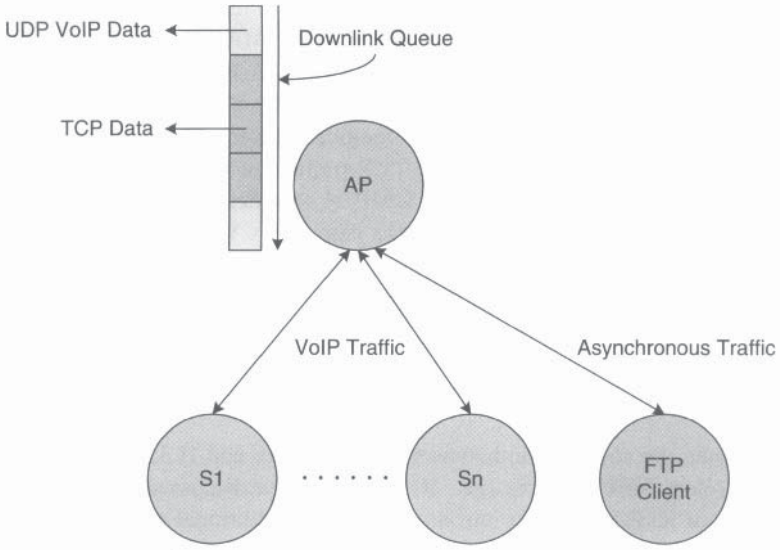


Figure 22.8 VoIP over WLAN with FTP traffic.

As a result, the bursty TCP traffic shares the same queue with UDP (VoIP) traffic. The main behavior of TCP traffic is that it aggressively sends out packets until packet loss occurs. As a result, TCP traffic effectively delays the channel access of VoIP traffic at the access point. A typical solution to this problem is the use of PQ (Priority Queue) in the access point with a higher priority given to the time-sensitive VoIP traffic. Using this method, the access delay can be effectively reduced from around 80ms to less than 5ms in the case of six VoIP streams and one FTP traffic.

On the other hand, the uplink TCP traffic (e.g., the TCP ACK for each TCP packet received) can greatly affect the VoIP traffic as well, especially in the case of the M-M scheme. The reason is that the M-M scheme takes advantage of the multicast nature of the wireless medium. However, multicast does not require any ACK for each packet received. Therefore, there is no mechanism to recover lost packets due to collisions. As a result, a multicast scheme is very sensitive to collisions. On the other hand, the uplink TCP traffic greatly increases the chance of collisions since it waits the same amount of time (DIFS) with other stations. In this case, with a large amount of TCP packets received, there is a large amount of uplink acknowledgement packets, or vice versa if the FTP transmission is in the reverse direction. Subsequently, a large amount of collisions with VoIP streams will be resulted. Using the M-M scheme or M-M scheme with PQ, and in the coexistence of six VoIP streams and one TCP stream, the packet loss rate can be more than 10%.

In view of the above problems, a modification to the MAC layer is proposed [345]. This involves the use of a different length of IFS (Interframe Space). Recall

from Section 9.4 that the shorter the length of IFS, the higher the priority to access the medium among different stations. As a result, a MAC-layer multicast priority scheme (MMP) with MIFS, which has a length longer than SIFS but shorter than DIFS, is proposed [345]. With the use of MIFS as the wait time for time-sensitive traffic, priority channel access is given to multicast traffic. As a result, the problem of contention between VoIP traffic and TCP traffic can be solved with the use of different priority wait time. Under the MMP M-M scheme, the loss rate is around zero but the access delay is increased in the case of six VoIP streams and one TCP stream. However, with the use of the M-M scheme together with the two solutions proposed—PQ and MMP, both access delay and loss rate are reduced.

22.6 SUMMARY

In this chapter, the cooperation between GSM network and H.323 VoIP network to support wireless VoIP services (i.e., iGSM) has been discussed. First of all, the architecture of iGSM has been introduced with the additional components that are required to support the services. Afterwards, various important issues in supporting the services have been discussed, including registration procedures, deregistration procedures, call-setup procedures, and some of the problems and their corresponding proposed solutions in using these services such as the tromboning effect and the misrouting problem.

Besides the iGSM system, as practical illustrations, we have also discussed VoIP over WLAN. Various issues on WLAN wireless technologies in supporting VoIP have been discussed, including the low bandwidth capacity of WLAN, poor performance in the presence of other data traffic such as FTP traffic, etc.

PROBLEMS

22.1 Briefly discuss the idea of iGSM in integrating the GSM network and H.323 IP network. Explain why additional components are necessary.

22.2 In iGSM, why there are the tromboning effect and misrouting problem?

22.3 Briefly discuss the idea of the M-M (Multiplex-Multicast) scheme in VoIP over WLAN and explain how it can improve the performance of VoIP over WLAN.

22.4 Why data traffic can greatly affect the performance of VoIP over WLAN? How can this problem be solved with the use of priority queue and MAC-layer multicast priority (MMP) scheme?

CHAPTER 23

WIRELESS VIDEO

> Don't part with your illusions. When they are gone you may still exist but you have ceased to live.
>
> —Mark Twain

23.1 INTRODUCTION

Video has become a commodity application in the wired Internet environment. Users routinely play back video files using their computers instead of using traditional audio/video equipment. Naturally, people expect the same to happen in a wireless environment—download video files wirelessly and play them back anywhere anytime.

Interoperability:

A wireless environment is known to have lossy channel with high error rate. This makes it difficult to provide real-time services in a wireless environment, where some QoS requirements have to be met. The real-time services envisioned are discussed in Section 23.2 while the system constraints are discussed in Section 23.3. With

Wireless Internet and Mobile Computing. By Yu-Kwong Ricky Kwok and Vincent K. N. Lau

Copyright © 2007 John Wiley & Sons, Inc.

the system constraints, different error concealment techniques have been suggested in order to tackle the issue of wireless channel. These techniques are introduced in Section 23.4. Error concealment techniques can be divided into three different categories based on the information hiding side, including forward techniques, post-processing techniques and interactive techniques. These techniques are discussed in Sections 23.4.1, 23.4.2 and 23.4.3, respectively. On the other hand, there are techniques which consider the effect of source coding and channel coding based on the varying wireless channel environment. This kind of joint source channel coding techniques are discussed in Section 23.5.

Performance:

On evaluating the performance of a wireless video system, we can adopt metrics like frame rate, bandwidth, delay and corrupted blocks. However, these metrics cannot effectively represent the video quality perceived by the viewer. The reason is that the perception of quality is very subjective, and therefore it is difficult to develop a satisfactory metric to evaluate this. Despite this, it is common to use average PSNR (Peak Signal to Noise Ratio) to evaluate the quality of a decoded video. The main aim of using average PSNR is to identify the maximum signal energy to noise energy ratio, which can be used to approximate the distortion of the decoded video.

23.2 REAL-TIME SERVICE GOALS

Real-time services involve service that requires the sending of data and/or video/audio between two or more parties. This kind of service is usually delay-intolerant. According to the delay-intolerant level accepted by applications, there are two main kinds of real-time applications.

The first kind is real-time interactive multimedia applications [285], such as peer-to-peer phone (VoIP), videoconferencing, etc. The aim of this kind of applications is to enable different users, who may have different communication devices, to enjoy multipoint communication with acceptable visual and audio quality. In addition, it is generally more tolerable to have relatively less quality in visual, but a relatively high quality of audio is preferred. Specifically, the audio service provided should be comparable to that provided by wired telecommunication service like PSTN. Consequently, in order to provide satisfactory quality of interactive service, a number of constraints like delay constraints and bandwidth constraints have to be considered. These system constraints will be further discussed in Section 23.3.

The second kind is streaming live multimedia applications [285], including live television programme, live radio program, etc. Strictly speaking, this kind is not exactly real-time service as it can tolerate a higher level of delay than that of the first kind. This kind of applications usually involves one-way sending of some video or audio from servers to multiple subscribed clients. Since it is relatively more delay-tolerant than interactive service of the first kind, its main aim is to provide clients with

acceptable quality of video and audio under loose delay constraints and bandwidth constraints.

23.3 SYSTEM CONSTRAINTS

Under the service goals discussed in the previous section, it is expected that certain minimum levels of network delay and bandwidth are required for real-time services so as to guarantee the quality of audio and video perceived by a user. In the conventional wired circuit-switched network or virtual circuit-switched network (e.g., PSTN), this can be done by resource reservation. However, nowadays, packet-switched network becomes a trend for nearly all kinds of networks, where guarantee on certain minimum levels of network delay and bandwidth is more difficult. The situation is worsen in a wireless network environment, in which an unreliable transmission medium is used. Specifically, the wireless medium is shared by all users, with the properties of having limited bandwidth frequent errors [285, 348].

Another problem is the possibility that different users are using different communication devices of different capabilities, and the devices are connected to different networks. It is expected that users may have real-time interactions through Internet connected by different access networks, for example, wireless networks, cellular networks, and ADSL, etc. The user devices and access networks would have very different capabilities in terms of computational power and bandwidth. Due to this reason, for real-time interactive communication between a 3G mobile phone user and a computer user with broadband Internet access, the heterogeneity between communicating parties have to be solved.

23.4 ERROR CONCEALMENT TECHNIQUES

In order to tackle the system constraints mentioned before, some techniques are required to detect and conceal errors generated. Generally speaking, there are three main types of error concealment techniques, as illustrated in Figure 23.1 [348].

The first type of error concealment techniques, i.e., forward techniques, is mainly initiated by the source. The source intentionally uses some schemes to hide errors from the destination. The most common examples of this type are layered coding techniques and transport level techniques, and the joint source channel coding techniques. In the second type, i.e., postprocessing techniques, the destination tries to hide errors from users by making use of two common techniques, namely spatial and temporal interpolation. The third one usually involves the cooperation between the source and destination. This type is called interactive techniques, and the examples are selective coding and retransmission techniques.

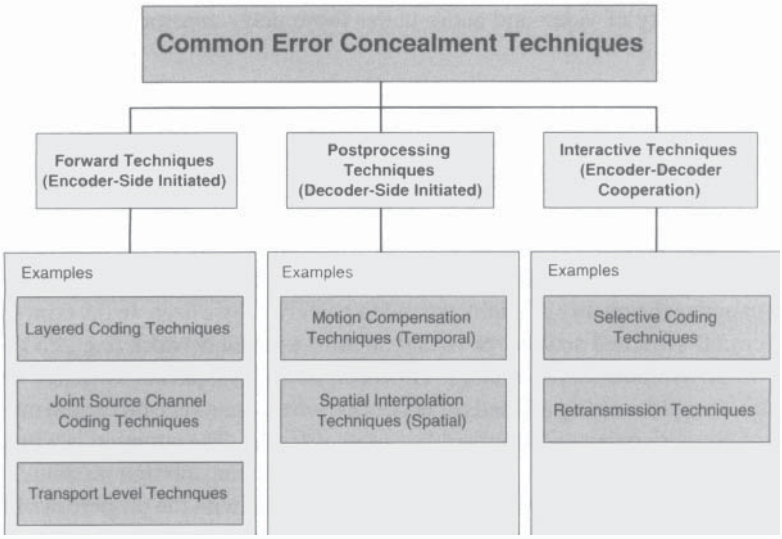


Figure 23.1 Examples of error concealment techniques.

23.4.1 Forward Techniques

In forward techniques, the source encoder tries to use some techniques to hide errors from the destination decoder. Several common techniques are discussed in the following.

23.4.1.1 Layered Coding Techniques Layered coding techniques are one of the most common example of forward error concealment techniques. The idea is summarized in Figure 23.2 [348]. In the figure, the source takes a raw video as the input. Then it encodes the raw video into multiple layers—the base layer (BL) and multiple enhancement layers (ELs). The BL usually involves the minimal encoded information which can be played out individually. Therefore, the BL defines the minimal bandwidth and delay constraints of the video, and is the most important layer. The EL involves the additional encoded information which can be used to refine the quality of the video. In order to provide multiple qualities of the video, several enhancement layers can be built.

Afterwards, the encoded BL and EL bitstreams are packetized and sent to the destination. Based on the bandwidth of the destination, the user can then select the layers that he is planning to receive in order to balance the tradeoff between quality and bandwidth available. As shown in Figure 23.2, the user can enjoy the minimal decoded video quality with the receipt of BL. However, if he can further receive the next enhancement layer, the decoded video quality can be refined by combining the BL and EL.

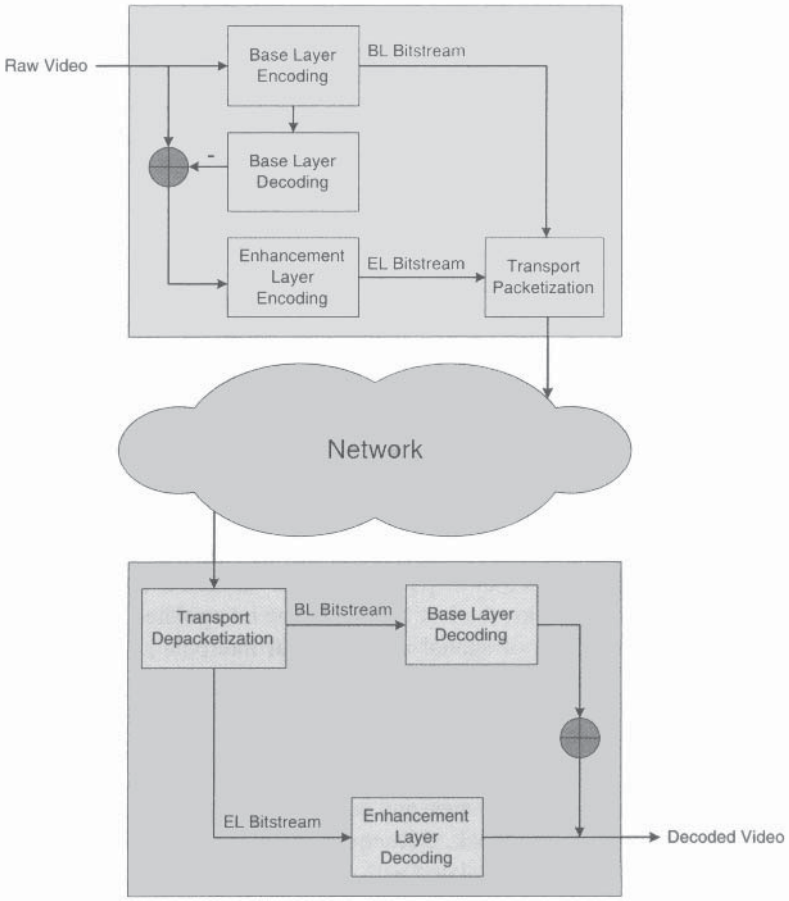


Figure 23.2 Layered coding scenario.

23.4.1.2 Transport Level Techniques There are several kinds of techniques that can be regarded as transport level techniques. The first one introduced in this section, i.e., prioritized packetization of layered coding, is in fact the continuation of the layered coding techniques in the previous section.

Figure 23.3 illustrates the idea of prioritized packetization of layered coding. In layered coding, it comprises the source coding of multiple layers involving the BL and multiple ELs. In fact, the receipt of BL is the most important as it is the basis of the quality of the video, while the receipt of the ELs further enhances the video quality. Due to this reason, the packets of the BL bitstream should be assigned with a higher priority than the EL bitstream, as illustrated in Figure 23.3 In general, packets with a higher priority will experience less packet loss and lower delay if the network in-between the source and destination is QoS-aware, for example, diffserv or intserv [348].

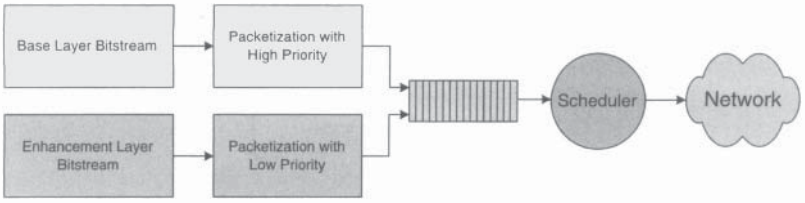


Figure 23.3 Prioritized packetization for layered coding.

Other than prioritized packetization of layered coding, some other techniques like interleaving can also be applied to help hiding errors. In fact, if the loss of packets includes non-successive blocks, the lost blocks can be interpolated with a relatively high accuracy with the help of spatial and temporal interpolation, which will be discussed in Section 23.4.2. Because of this reason, the idea of packetization of odd-even blocks are suggested [373, 348]. As shown in Figure 23.4, the blocks of a frame are marked with odd and even numbers. During packetization, the even blocks from the same slice of a macroblock are filled in first, followed by the odd blocks. By doing so, if there is packet loss, the even and odd blocks in the macroblock of a frame are referred to identify the lost block. Subsequently, by using the correctly-received blocks the lost block can be interpolated with a relatively high accuracy.

In addition to the interleaving techniques mentioned above, cell level forward error correction (FEC) codes like Reed-Solomon codes are commonly applied to conceal errors caused by cell loss [348].

23.4.2 Postprocessing Techniques

The destination side is usually involved in postprocessing techniques to figure out the occurrence of errors due to various reasons and to conceal the errors from the user. If there are missing packets, since the destination side decoder does not have correct

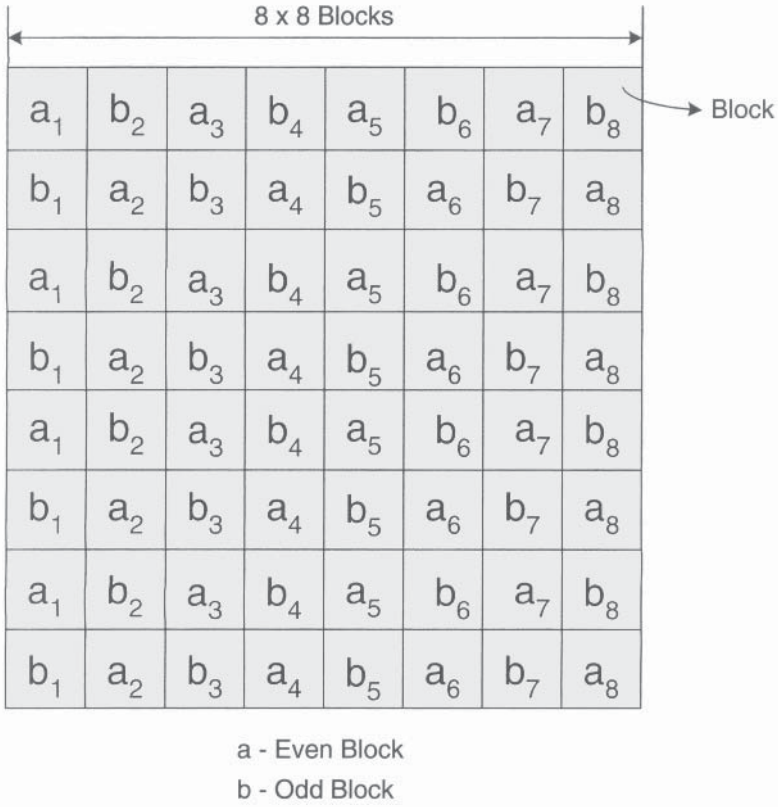


Figure 23.4 Block interleaving.

encoded information, then the corresponding macroblocks will be affected. However, the missing information can be fabricated by referencing the correct information of neighboring blocks in either time or spatial domain. In other words, postprocessing techniques involve the use of interpolation in time domain and spatial domain as illustrated in Figure 23.5 In fact, as for interpolation, some simple techniques can be applied, such as replacing the missing macroblocks from neighboring macroblocks of the same frame or replacing the missing macroblocks from the macroblocks of previous frame. However, these techniques will result in strange images which may be noticeable by the user. This will be further discussed in subsequent sections.

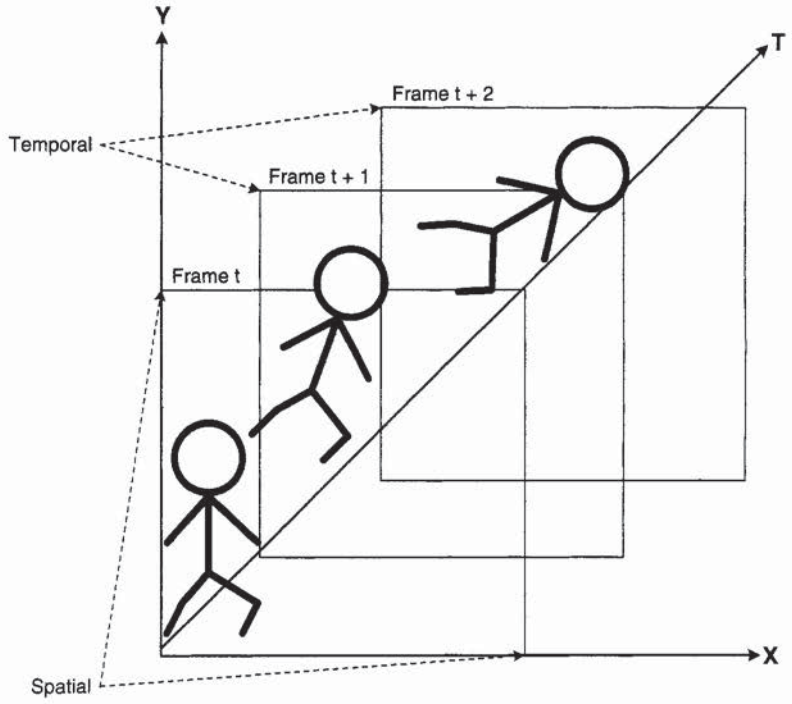


Figure 23.5 The idea of postprocessing techniques.

23.4.2.1 Spatial Interpolation Techniques Spatial interpolation techniques are proposed mainly based on the idea that neighboring macroblocks within the same frame should be very similar to each other. In this section, two spatial interpolation techniques will be discussed.

The first one is block-based spatial interpolation. As depicted in Figure 23.6, a unit square represents a pixel while a square of $8x8$ pixels represents a macroblock. In addition to macroblocks, a square of NxN pixels is defined as a block (in the figure, $N = 4$). Suppose that the macroblock inside the figure is missing, then the pixels

inside the missing macroblock are fabricated by interpolating the two nearest pixels from two other correctly-received neighboring blocks. As illustrated in Figure 23.6, the pixel currently in consideration has a distance d_T from the upper block and a distance d_L from the left block. The interpolation is then performed based on these two pixels with distance being a weight [13, 348].

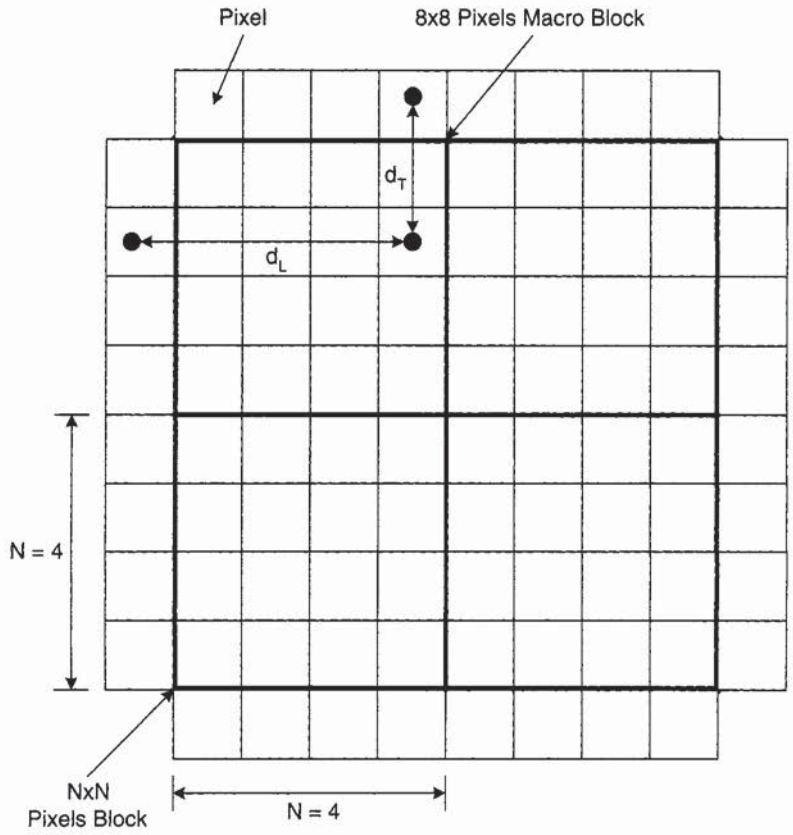


Figure 23.6 Block-based spatial interpolation.

The second one is macroblock-based interpolation. As shown in Figure 23.7, suppose that we are considering the same missing pixel inside the same macroblock as in Figure 23.6, the missing pixel this time is fabricated by interpolating the four nearest pixels from the four neighboring macroblocks. In this case, the pixel in consideration is interpolated from four nearest pixels with distances d_T, d_B, d_L and d_R. Again, the interpolation is performed with distance being a weight.

23.4.2.2 Temporal Interpolation Techniques Temporal techniques are mainly based on the idea that two macroblocks in the similar region should be very similar

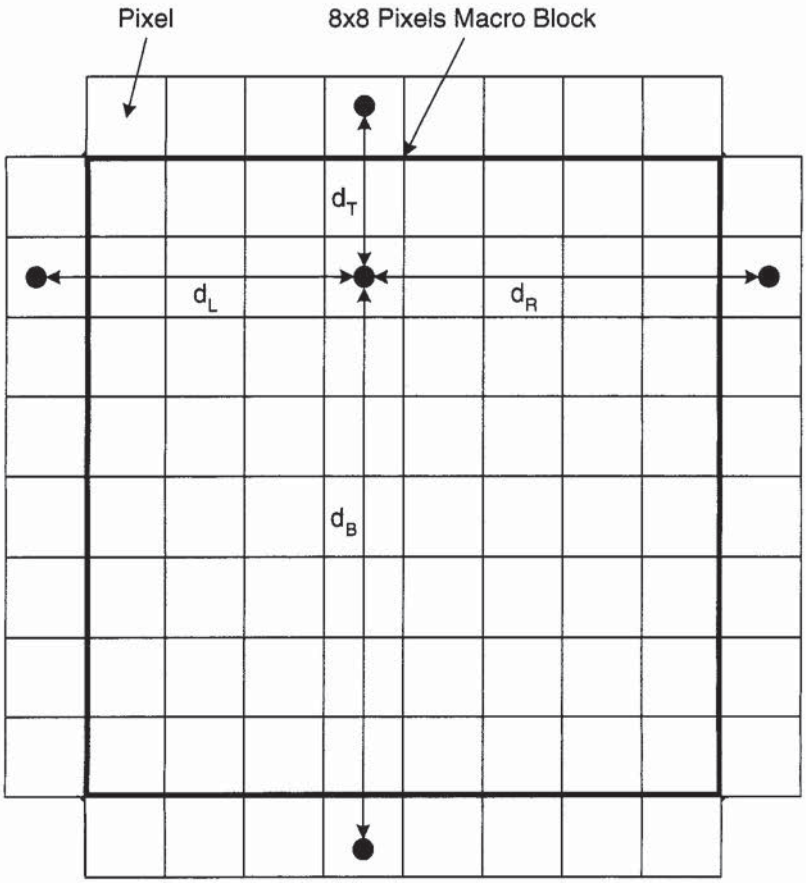


Figure 23.7 Macroblock-based spatial interpolation.

in two consecutive frames. However, this idea is correct for still consecutive frames. If the objects are in movement, there will be differences in two consecutive frames. In this case, directly applying the macroblocks of the previous frame to the missing macroblocks of current frame will result in a strange image noticeable by the user. Consequently, as shown in Figure 23.8, if the object is in motion, motion vectors of the previous frame and current frame are generated and sent to the receiver. During decoding process, the motion vectors are considered in regenerating the images in the frames. If certain macroblocks are missing as shown in the figure, then the motion vector of the missing block is interpolated from the motion vectors of the neighboring macroblocks. This scheme is based on the idea that objects, which usually occupy several macroblocks, will move in the same direction in consecutive frames. With the motion vectors of the missing macroblocks, the missing macroblocks can then be regenerated by either selecting the appropriate macroblocks from the previous frame in replacement of the missing one or interpolating the macroblocks of the previous frame based on the motion vectors. Schemes that make use of this idea are also known as motion compensated concealment schemes [79, 348].

23.4.3 Interactive Techniques

Interactive techniques usually involve the cooperation between the sender and the receiver. Upon discovering the occurrence of some error, the receiver will actively contact the sender, and the sender will respond by sending the original copy of the error packet (retransmission) or changing the encoding of subsequent frames to remove the dependency on the error packet (selective encoding).

23.4.3.1 Retransmission Techniques When the destination side discovers that there are errors, it will then contact the source about the error region. In some streaming applications as discussed in Section 23.2, it is affordable for the destination to wait for the retransmission packets as there is usually certain amount of play out delay for the buffered packets. However, for real-time applications, retransmission waiting time may not be affordable. In this case, before the retransmitted packets arrive, the destination side can use the error concealment approaches discussed before to conceal the error while waiting for retransmission. At the same time, subsequent frames and blocks depending on the missing blocks are marked. In this case, when the retransmission packets arrive before the play out time of the affected frames and blocks, those affected blocks are then reformed [348].

Besides the stopping of error cascading effect of a missing block, retransmission can also be done based on the priority of the layers as discussed in Section 23.4.1. Recall that the loss of packets in the BL bitstream is much more intolerable than that in the EL bitstream. In this case, multiple copies retransmission can be applied if the high priority packets are lost.

23.4.3.2 Selective Encoding Specifically, one of the most important effects of retransmission of packets as mentioned above is to stop error propagation due

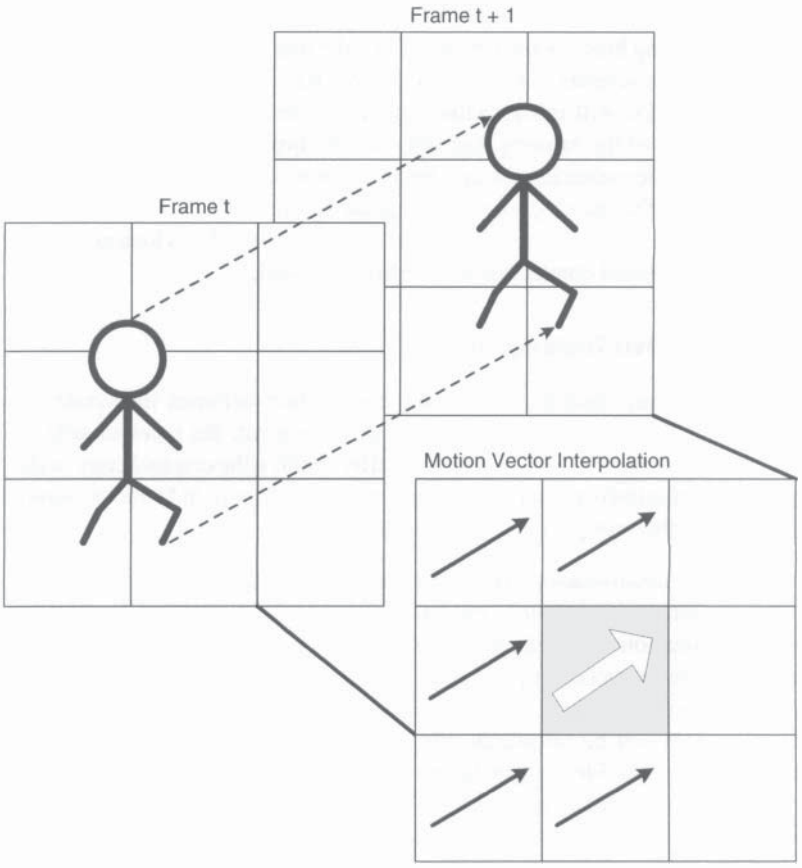


Figure 23.8 Motion compensation temporal interpolation.

to missing or error blocks. In addition to retransmission, there are also some other approaches. The simplest approach is that the receiver informs the sender about which block in which frame is in error. The sender will then encode the next frame into intracode mode. However, under this scheme, there is no dependency on the previous frames, and the compression efficiency will be greatly reduced due to the fact that techniques like motion vector cannot be applied to reduce the information transmitted [348]. To mitigate this problem, selective encoding is introduced. The main idea of selective encoding is that whenever the receiver detects erroneous blocks, it reports the error to the sender. Then the sender can continue to encode the video with dependency of the missing blocks up to current frame and the affected area is also calculated. On the other hand, the next frame is encoded without the dependency of the previous missing blocks or areas.

23.5 JOINT SOURCE CHANNEL CODING TECHNIQUES

In a wireless video scenario as depicted in Figure 23.5, the raw video is first encoded by the source coder, which aims at reducing the redundancy in the raw data, and then further encoded by the channel coder, which aims at protecting data from channel specific errors. Traditionally, source coders and channel coders are designed separately [348, 285].

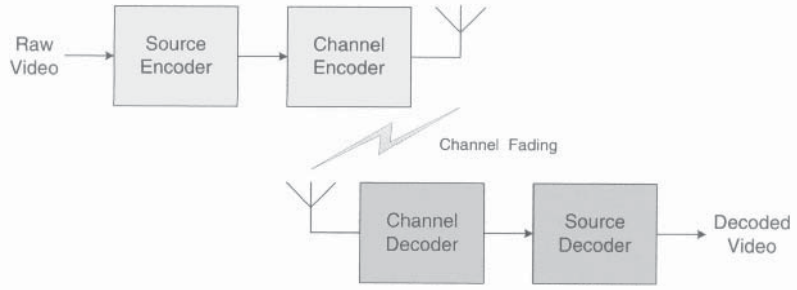


Figure 23.9 Source and channel encoders and decoders.

Shannon suggested that the source coder and channel coder can be designed separately to achieve an optimal performance in the case that the source coder is an optimal one which produces data rates lower than that of channel capacity. At the same time, the channel coder is also an optimal one. However, it has been suggested that Shannon's idea is impractical to be achieved with reasonable design complexity of the coder, especially in a wireless environment where there is channel fading causing various channel errors. In order to solve this problem, joint source channel coding is developed [98].

In fact, under noisy channel, increasing the source coding rate does not necessarily result in a better quality of video due to the reason that less bandwidth will be available

for channel coding. As a result, on designing joint source channel coders, a trade-off has to be balanced between the effects of source coder and channel coder under certain fading channel (e.g., Rayleigh fading channel).

Many approaches have been proposed in joint source channel coding. One of the typical examples is the use of convolution code in protecting the MSB (most significant bit) generated by source coder DPCM (differential pulse code modulation) [245] and DCT (discrete cosine transform) [246]. The idea is then extended to the evaluation of the importance of certain bits to the video quality and different coding rates are used to protect the bits from channel errors.

23.6 PRACTICAL ILLUSTRATIONS—RATE ADAPTATION FOR MPEG-4 VIDEO

In previous sections, various error concealment techniques have been introduced. In this section, we present a practical illustration which combines the techniques introduced before to improve the decoded video quality of MPEG-4.

The rate adaptation for MPEG-4 video [319, 370] involves three main components as shown in Figure 23.10. The three components are:

- Rate Adaptation of Scalable Coding
- Prioritized Packetization
- Differentiated Forwarding

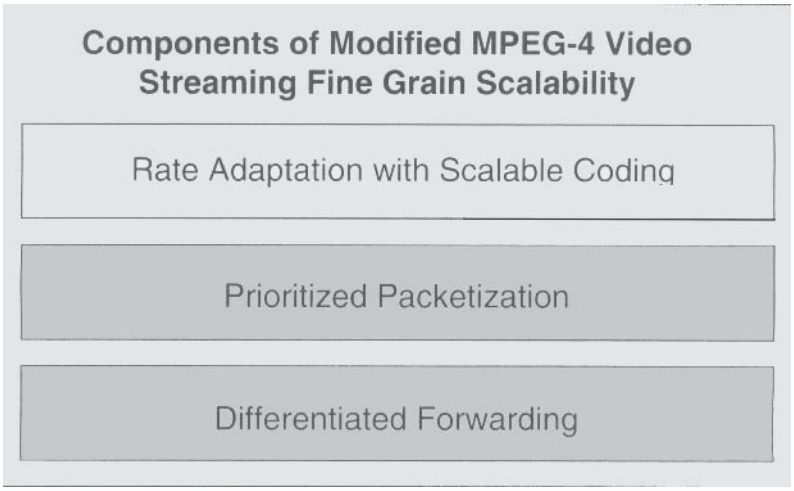


Figure 23.10 Components of rate adaptation for MPEG-4 video streaming.

Figure 23.11 [319, 370] illustrates the framework of the rate adaptation MPEG-4 FGS (fine grain scalability) with prioritized packetization and differential forwarding. In MPEG-4 FGS, the video stream can be encoded into two layers—base layer (BL) and enhancement layer (EL). Unlike the idea of layered coding in Section 23.4.1, the enhancement layer encoding in MPEG-4 FGS is more flexible with the use of bitplane coding in which each bitplane can be adjusted independently based on the bandwidth requirement, i.e., the receipt of certain bitplane of certain rate can increase the video quality in MPEG-4 FGS instead of the need for successful receipt of the whole EL. The BL defines the minimum bandwidth requirements for the decoding of the video and so the bandwidth for the BL bitstream can define the minimum bandwidth requirement of the network. On the other hand, rate-distortion information can be calculated and embedded into each bitplane independently by considering the first several samples because the same bitplane has similar rate-distortion characteristics. The rate for EL bitplanes can then be selected based on the estimation of the current bandwidth so as to maintain a relative constant video quality (minimal distortion variance between different frames of the video). This idea is known as rate adaptation of MPEG-4 FSG.

The rate-adapted BL and EL bitstreams are then packetized and assigned with a priority based on its relative impact on the video quality with the loss of packets (including the impact on current frame and subsequent frames). This idea is known as prioritized packetization.

Afterwards, the prioritized packets are mapped into different QoS queues in the assured forwarding in diffserv, which has different QoS agreements in terms of bandwidth and delay and/or packet dropping rate. Generally speaking, BL packets are assigned to a higher priority queue due to the fact that the BL defines the minimum bandwidth requirements and has relatively less flexible adjustment. The priority assigned inside the packets can then be used to determine which packets to be dropped inside the same queue, so as to minimize the impact on the video quality. This idea is known as differentiated forwarding.

The combination of the three components aims at maintaining a relatively constant video quality even if there are fluctuations in bandwidth availability, with the help of rate adaptation based on the network estimation performed by either the source or destination side. This is illustrated in Figure 23.11

23.7 SUMMARY

This chapter begins by discussing the real-time service goals and system constraints of a wireless video system. There are two main kinds of real-time services expected. The first one is real-time interactive services like videoconferencing and the second one is streaming live multimedia applications. These two kinds of services have different requirements in terms of network bandwidth and delay.

The lossy link and low bandwidth characteristics of wireless medium result in high error rate in packet transmission. As a result, some error concealment techniques are

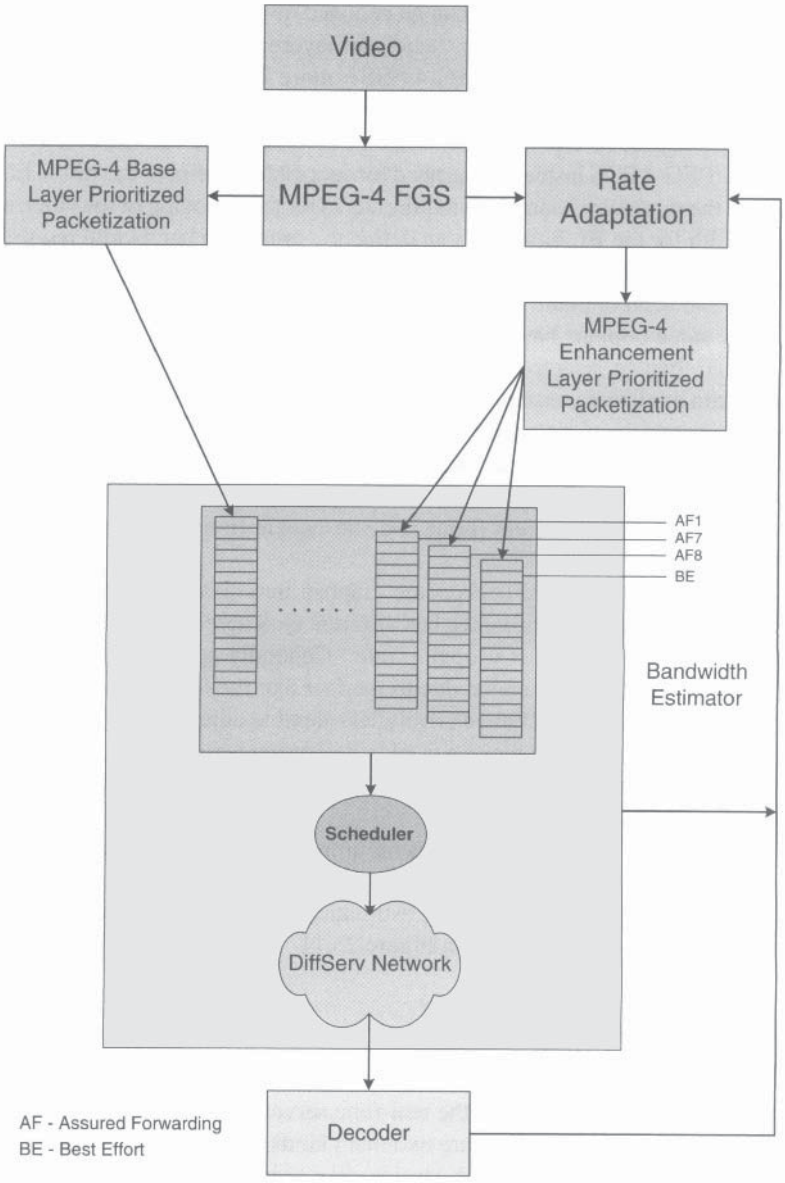


Figure 23.11 Rate adaptation for MPEG-4 video streaming framework.

required to conceal errors from the user. There are three main categories of error concealment schemes—forward, postprocessing and interactive.

Forward techniques usually involve the source which uses some measures to help hiding errors. For example, in layered coding, it encodes the video into BL and EL bitstreams, while the transport level techniques include prioritized packetization of layered coding, block interleaving, and cell-level FEC, etc. Postprocessing techniques usually involve the destination side which uses some measures to help hiding errors from the user. The most common techniques in postprocessing involve spatial and temporal interpolation. In spatial interpolation, it takes advantage of similarities of neighboring macroblocks in the same frame. In temporal interpolation, it takes advantage of similarities of macroblocks of the similar region in consecutive frames. Finally, interactive techniques involve the cooperation between the source and destination. Specifically, the destination side actively informs the source side about the occurrence of errors. Afterwards, the source will try to stop error propagation by either retransmission or selective encoding, so as to remove adverse effects of erroneous macroblocks to later frames.

Lastly, a practical illustration which presents how the error concealment techniques introduced in this chapter are combined in MPEG-4 FGS with rate adaptation, prioritized packetization and differentiated forwarding.

PROBLEMS

23.1 In this chapter, several error concealment approaches are introduced. Briefly discuss these approaches and their pros and cons.

23.2 Briefly discuss the joint source channel coding approach and its pros and cons.

This Page Intentionally Left Blank

CHAPTER 24

WIRELESS FILE SYSTEMS

> Information appears to stew out of me naturally, like the precious ottar of roses out of the otter.
>
> —S. L. Clemens (Mark Twain): Roughing It, pref., 1872

24.1 INTRODUCTION

File system is a fundamental, if not the most basic, service in any computing environment. The "file" abstraction matches human's natural intuition of organizing information. It is mandatory for a wireless system to provide a convenient file system.

Interoperability:

Wireless and wireline file systems have quite a number of differences. Generally speaking, a wireline file system can be assumed to be reachable all the time. As a result, certain modification on the file system can be broadcast to all of the servers at the same time. In order to improve the availability of files, replication of files on multiple servers connected through a wireline environment can be adopted. This

Wireless Internet and Mobile Computing. By Yu-Kwong Ricky Kwok and Vincent K. N. Lau
Copyright © 2007 John Wiley & Sons, Inc.

can ensure that files accessed by clients are the most updated ones. However, when certain files are being modified by some clients, other clients might not be able to access the files in order to prevent conflicts. This is discussed in Section 24.4.

On the other hand, in a wireless environment, the assumption of always online may not be correct. Due to some bandwidth limitations such as reduced throughput rate and increased data transfer cost, mobile devices might not be able to connect to the servers at all or they only have some limited connections to the servers. To solve the problem, disconnected operations and weakly connected operations are introduced, which are discussed in detail in Section 24.5 and Section 24.6, respectively. These two kinds of operations can help improve the availability of files with the use of some techniques like hoarding, emulating, reintegration, etc.

Both replication method and disconnected and weakly connected operations can help improve the availability of files in a wireless and distributed environment. However, they target on different situations. Replication method targets more on consistency, while disconnected and weakly connected operations target more on availability. This leads to an interoperability problem as the two kinds are considered to be competing with each other in nature. However, the idea of file replication through multiple connected servers can, in fact, be combined with the idea of disconnected and weakly connected operations, so as to get the best of both worlds. As a result, the two kinds can also be complementary to each other, by taking special care of interoperability issues mentioned before. In this chapter, the Coda file system is discussed in detail to demonstrate the possibility of tackling the interoperability issues.

Performance:

File system performance is traditionally quantified by data response time and data transfer throughput. In a distributed file system, the control overhead needed to maintain data consistency is also an important measure. These metrics are also commonly used in assessing the performance of a wireless file system.

24.2 FILE SYSTEM SERVICE MODEL

Coda is a distributed file system designed to support mobile file systems with the support of several novel operations, including optimistic replication in server replications, disconnected and weakly connected operations, etc. With the help of these operations, mobile file system support becomes possible.

Figure 24.1 illustrates the service model of the Coda file system [310]. In the figure, there are two sides involved—client side and server side. In the client side, client $1a$ and client $1b$ represent the client under strongly network-connected situation. In this mode of operation, the client can read files from the server and make updates to the server immediately. On the other hand, client 2 represents the weakly connected mode of operation. In this mode of operation, the client may have limited bandwidth connection to the servers, e.g., a mobile client updates through wireless connections. Lastly, client 3 represents the disconnected mode of operation. In this mode of

operation, the client may not be able to reach the servers to get the most updated items.

On the other hand, server $1a$ to server $1c$ represent the servers under normal mode of operation. As a result, any update on one server can be propagated to others immediately. Server 2 represents a server which is isolated from the connected server group. This is possible due to specific server failures or network disconnection between two groups of servers. In this case, the servers need to resolve conflicts after reconnection of the failed server.

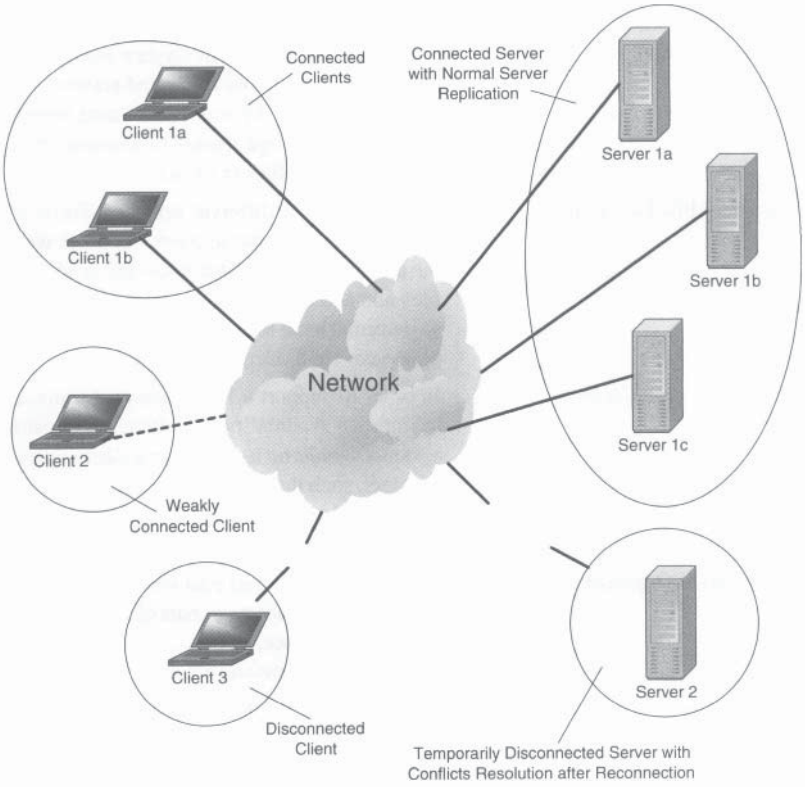


Figure 24.1 Coda file system service model.

With the help of the service model just discussed, mobile clients with different network connection modes (i.e., normal network connection, no network connection and limited network connection) can access to a mobile distributed file system. In the following sections, we discuss how Coda copes with the characteristics of different modes of operation without greatly affecting the services provided to users in other modes.

24.3 GENERAL PRINCIPLES FOR THE DESIGN OF A DISTRIBUTED FILE SYSTEM

In the design of a distributed file system that can support the service model mentioned before, there are a number of considerations. Some of the major ones are summarized in Table 24.1 [310].

Table 24.1 Principles for the design of a distributed file system.

Principle	Description
Client-Server Model	Generally, a distributed file system is expected to have a limited number of trusted servers which can replicate user files and serve user requests, and a relatively large amount of untrusted clients that may access files anywhere.
Transparent File Location	The identities of different servers holding files should be transparent to users. In other words, the users should be able to know the existence of files and access them through local name space as if normal local file access. The contact details of servers are hidden from users.
Local File System Caching	In order to support a large mount of users, and to improve availability and efficiency, local file systems should be used to cache files requested by users, such that operations on the files can be performed locally and modifications are updated back to the servers afterwards.
Volume Based Organization of Data	Data files are grouped into volumes. Each volume refers to a subtree or part of s subtree of the whole name space. This kind of organization can improve administrative efficiency.
Access Control	Authentication mechanisms should be adopted to identify users. Access control mechanisms should also be used to control the access rights of different users.

24.4 REPLICATION SERVICES AND MECHANISMS

The most basic idea for improving the availability of client files is to replicate files. The replication of client files is usually performed by various servers located in different places, and the servers replicate client files based on some predefined rules. As a result, files can be retrieved whenever a client is able to contact either one of the

servers with replicated files. This can eventually improve the availability and prevent single point of failure.

However, replication also causes some problems. For example, a write to the distributed file system is required to be propagated to all the servers holding the replica simultaneously. In addition, it is required to handle lots of conflicts like concurrent update by multiple clients on the same file. In order to reduce the complexity of replication, pessimistic replication is widely employed. With this method, access to a file may be blocked with the presence of network partition in order to preserve data consistency. However, this method is not scalable. As a result, optimistic replication is incorporated into the Coda file system [311, 310]. Using optimistic replication, read and write operations are permitted with network partitions as long as the servers within the partition maintain consistency. Conflicts are then left to be resolved until the network partitions are merged.

In Coda, the basic unit of replication is volume. A volume contains a tree structure in which the root directory of the volume, files and subdirectories below the root directory are present. Mobile clients view only a normal file system under the directory /coda. This abstraction is made possible by the Venus cache manager as shown in Figure 24.2. Venus is the cache manager located inside mobile clients and it contacts the appropriate Coda servers holding the required replicas whenever files under /coda are referenced inside the clients.

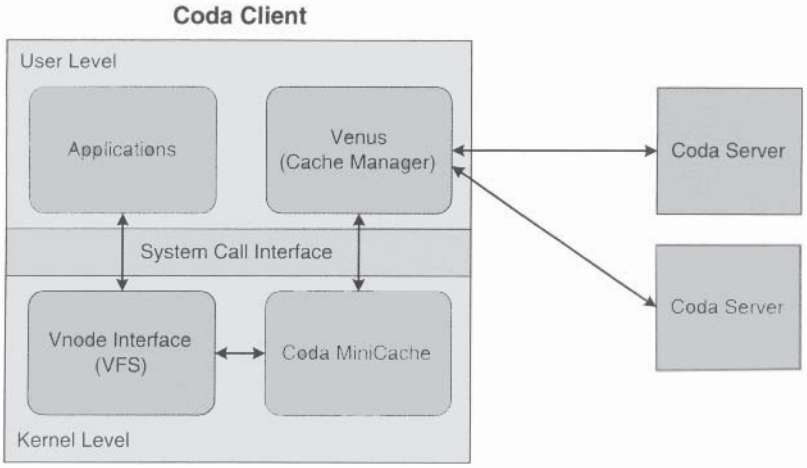


Figure 24.2 Coda client architecture.

The group of servers holding replicas of a particular volume forms the VSG (Volume Storage Group). However, due to network connectivity problems, not all the servers holding the replicas of the particular volume can be contacted. As a result, *the group of servers* which can be contacted at the time of access forms the AVSG

(Accessible Volume Storage Group). All the accesses including read and write are then be performed on the servers within the AVSG. In this case, it is possible that the size of AVSG is smaller than that of VSG due to some temporary network brokage. However, this optimistic replication approach may result in conflicts which are required to be resolved between servers within the whole VSG. This is in fact a tradeoff between scalability and consistency.

Figure 24.3 illustrates the idea of read operation of the Coda client. When the Coda client references files which are not found within the cache, Venus first randomly selects a server within the AVSG as a PS (Preferred Server). In Figure 24.3, server 1 is selected as PS. Venus then sends data read message with PS to all the servers in the AVSG. The server which is named as PS then returns the status information together with data required by the client back to Venus, while other servers also return the status information back to the client. Afterwards, Venus compares the information obtained by all the servers in AVSG. If there are mismatches in the status information, conflicts occur and have to be resolved. For example, the stale copy of the referenced data inside a server in the AVSG will be updated. After that, the read operation is restarted.

After discussing the read operation, we discuss the update operation. Figure 24.4 illustrates the update operation of the Coda client. When a certain file is modified, the it is sent to all the servers in the AVSG simultaneously. This operation involves two phrases, known as $COP1$ and $COP2$. $COP1$ includes the sending of updated operations such as creating a directory, transferring of the file involved, and/or changing the access lists. On the other hand, $COP2$ involves the process of informing the servers in the AVSG about the servers which have successfully performed $COP1$. This information is summarized in an update set message.

If no failure occurs during the update operation, all the members within the AVSG are synchronized. If a server or network failure occurs during the COP operation, the client times out and assumes that the servers which have not replied are not updated. At the same time, if a server times out, it is assumed that no other servers receive the $COP1$ message. In these scenarios, conflicts will be detected and resolved after the server or network is restored.

24.5 DISCONNECTED OPERATIONS AND CACHING

In the previous section, server replication has been discussed in detail. The growing popularity of portable devices has lead to another view of improving availability of files—disconnected operations [191, 310]. Portable devices usually have limited storage space. Therefore, users usually download files that can be fit into their portable devices and it is highly probable that the files will be used in the near future. Because of these reasons, in disconnected operations, the LRU (Least Recently Used) policy is utilized, which is assisted by the prediction of file access of users.

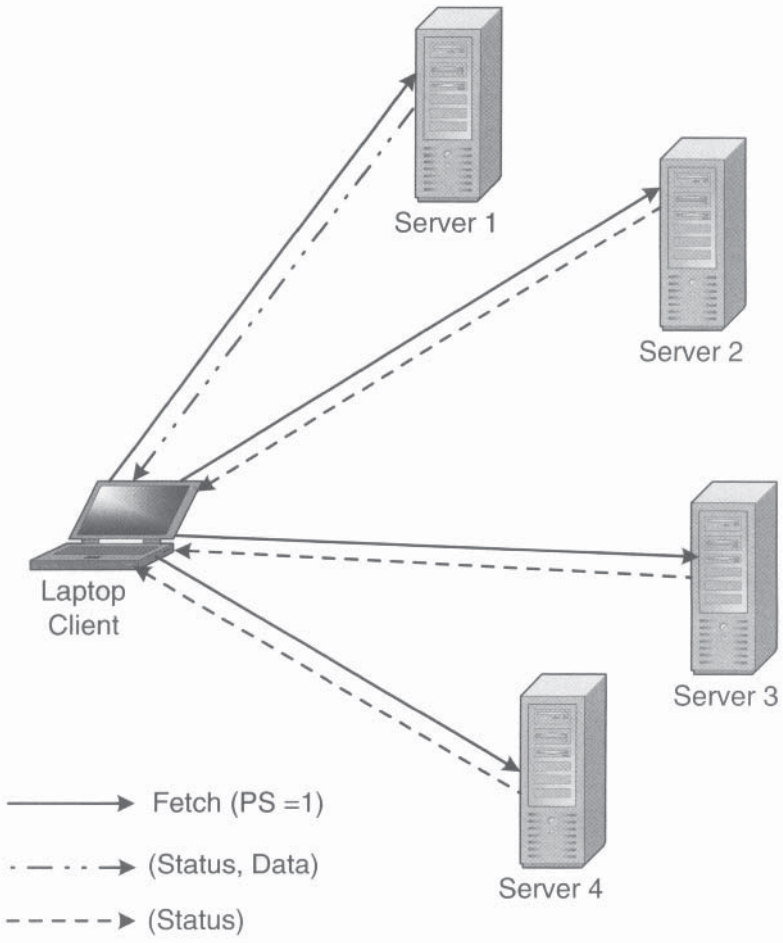


Figure 24.3 Coda server replication read process.

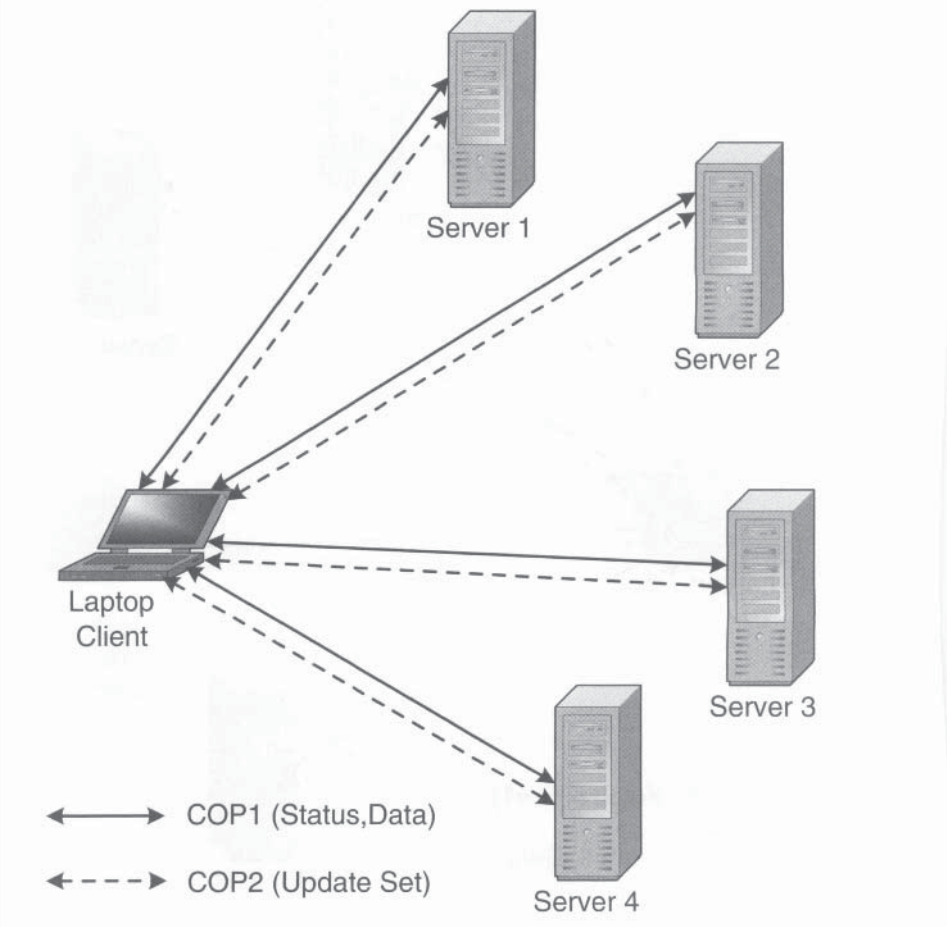


Figure 24.4 Coda server replication update process.

No matter whether it is due to voluntary or involuntary network disconnection, during disconnected operations, the same file system name can still be used. The files which have been downloaded can still be accessed and updated, and possible associated conflicts are automatically resolved after the client reconnects to the network.

Figure 24.5 illustrates the implementation of disconnected operations in Coda. In order to support disconnected operations, three states are defined in the Venus cache manager according to different situations—hoarding, emulating and reintegrating. In normal situation, Venus is in the hoarding state which performs read/write operations normally as discussed in the server replication part previously. However, when disconnection occurs, Venus then transits to the emulating state in which servers are pretended to be present. After reconnection, Venus transits to the integrating state in which all the updates and files are synchronized with the servers. After the synchronization process, Venus returns to the hoarding state. Table 24.2 further describes the characteristics of these states.

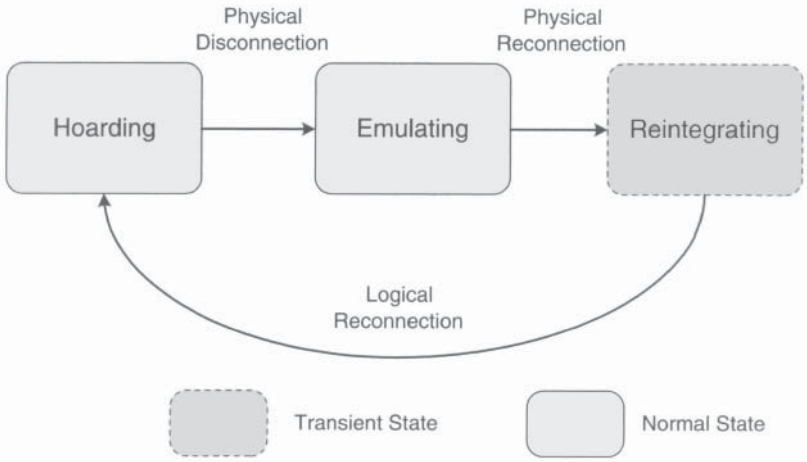


Figure 24.5 Disconnected operations in Coda.

After discussing the server replication method and disconnected operations in Coda, we can see that these two schemes in fact have competing roles in nature. That is, if disconnected operations are possible to solve the problem of availability, then it may not be justified to use server replication, which is generally more expensive. However, it is worthwhile to point out that server replication and disconnected operations also have complementary roles. The replicas in the server replication have higher quality in the sense that they are more complete and accurate. At the same time, users are able to access the files in different geographical areas. On the other hand, the cache in disconnected operations has lower quality, but it provides higher availability of files and performance. Due to the complementary role of these two

Table 24.2 Characteristics of different states in Coda disconnected operations.

States	Description
Hoarding	In this state, the client is always alert of disconnection by caching some files that are important. However, due to the limited storage space in the client device, only certain files can be cached. To choose the appropriate files to be cached, two kinds of information are used—implicit and explicit. Implicit information is based on algorithms like LRU as stated before. On the other hand, explicit information is based on the client hoard database (HDB), which includes the files that are explicitly required to be cached. With these two types of information, Venus periodically checks for the list of caching files to prepare for disconnection.
Emulating	Venus enters this state during disconnection. In this state, read/write requests on the Coda file system are redirected to the cache. However, only those requests which involve files in the cache can be satisfied. On the other hand, any changes to the files are required to logged, so that the updates performed during the period of disconnection can be delivered back to the system without loss. This is done by the client modification log (CML). In order to reduce the number of records on the log, certain optimization is performed on the log file. For example, if the user creates a file, saves it and then deletes it, there will be three actions recorded in the log. By performing log optimization, these three records are figured out and removed.
Reintegrating	During reintegration, with the help of the CML, Venus performs update operations as stated in the server replication part previously. The logged updates are propagated to the AVSG and consistency is checked by the servers in the AVSG.

mechanisms in improving the availability files, Coda is designed to integrate both mechanisms to get the best of both worlds.

24.6 WEAKLY CONNECTED OPERATIONS

As discussed before, disconnected operations are motivated by the popularity of portable devices. Nowadays, most of the portable devices are integrated with certain wireless technologies. For example, mobile phones are integrated with 2G/3G technologies, laptops and PDAs are integrated with IEEE 802.11 a/b/g, etc. On the other hand, since disconnected operations have relatively low quality of files due to the use of caching, researchers have started thinking about the possibility of using temporary wireless connections to improve the quality of the cache. This results in the emergence of weakly connected operations [249, 310].

Weakly connected operations are very similar to disconnected operations. However, as shown in Figure 24.6, there are some fundamental differences. In the design of weakly connected operations, Venus can enter three states—hoarding, emulating and write disconnected. Different from that of disconnected operations, the transient state, i.e., reintegrating, is replaced by the write disconnected state. Upon recovering from disconnection, Venus first enters the write disconnected state. It transits to the hoarding state only after all the updates are performed and a strong connection is detected.

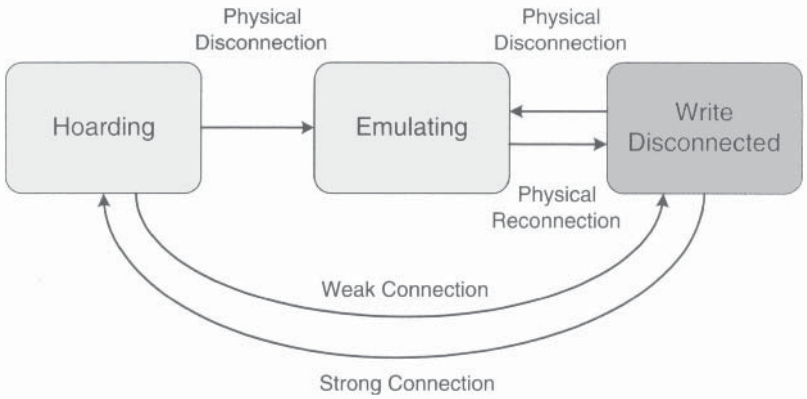


Figure 24.6 Weakly connected operations in Coda.

The write disconnected state is the state in which weakly connected operations differ from disconnected operations. This state is a mixed state between the hoarding, emulating, and reintegrating states. In the write disconnected state, updates are logged similar to that in the emulating state. At the same time, the updates are propagated to the servers using trickle reintegration asynchronously. On the other hand, since the device has limited connection ability in this state, cache misses which cannot be served in the emulating state can now be served. However, if the size of the file requested is too large, user intervention may be required.

As mentioned before, CML (client modification log) is the place for logging updates, and it is very useful for keeping it compact. Due to this reason, optimization techniques are usually applied to the CML as explained in the previous section. However, the use of trickle reintegration may result in the case that some updates are propagated to the servers before optimization can be performed on the CML. To mitigate this situation, an aging window is adopted into the log file to prevent premature updates from reducing the efficiency of log optimization.

Figure 24.7 shows a typical CML log file which is under trickle reintegration. The updates can only be propagated after a certain time governed by an aging window. Specifically, in the figure, only those updates older than time A can be propagated.

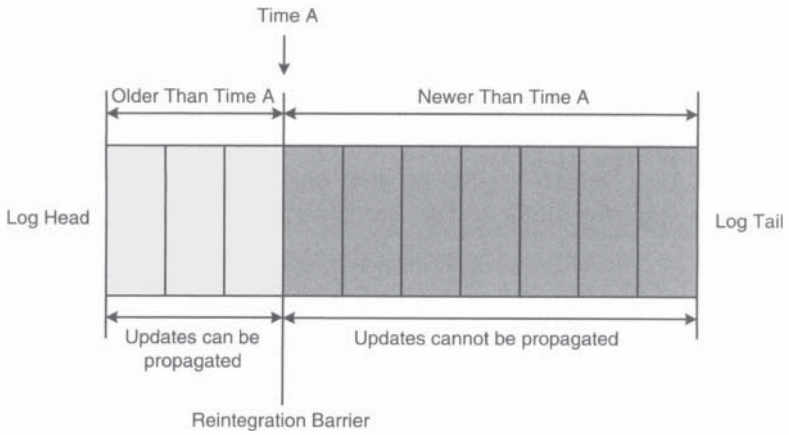


Figure 24.7 Typical CML in trickle reintegration.

Additionally, there is one more technique which can be used in trickle reintegration to improve the performance in the write disconnected state—the choosing of a different reintegration chunk size. This size can be chosen according to the connection link quality and bandwidth. As a result, different connection environments and situations can be covered with this technique.

24.7 PRACTICAL ILLUSTRATIONS—MOBILE DISTRIBUTED DATABASE

In previous sections, we have discussed the Coda file system in detail, especially the details on how it copes with mobile users with different network connection abilities and how it maintains the consistency of files in the system. The discussion focuses on mobile users with local file systems, such as laptop users, handheld device users, mobile phone users, etc. In this section, the practical implementation of a mobile distributed database will be discussed in detail. We discuss how the design of a distributed database can greatly improve the network loading of an existing GSM network [233]. Specifically, we focus on the backbone network of the wireless system.

Figure 24.8 shows the architecture of a GSM network. Within the GSM network, there are two main types of database—HLR and VLR. HLR is the database which stores the user information, including the current locations of users, user profiles, etc. On the other hand, VLR, which is usually located together with an MSC (Mobile Switching Center), serves as a temporary storage of user information within the specific location. Whenever a user moves from one VLR to another VLR, location update is carried out. This update involves the HLR as well. On the other hand,

whenever there is a call to a specific user, the user's HLR is queried also. As a result, loading on the HLR is in fact quite heavy.

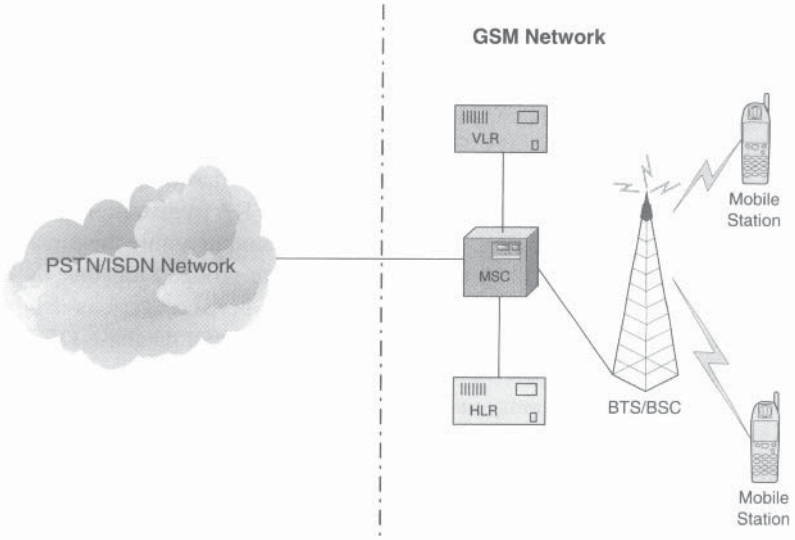


Figure 24.8 GSM database architecture.

A number of methods have been proposed to reduce the loading on the HLR. These methods include:

- Forwarding strategy. A pointer is set up from the old VLR to the new VLR without updating the HLR record.
- Anchoring strategy. A certain location is served as a local anchor and updates only affect this anchor. Moreover, the HLR always points to this anchor.
- Caching strategy. The location of a mobile terminal obtained from the previous call is cached in the signal transfer point (STP). When a call passes through the STP and targets for the specific mobile terminal with an entry inside the STP, the cache is used without querying the HLR.
- Replication strategy. The location of a mobile terminal is replicated in several selected databases. Afterwards, a call originating from the service area of these databases can obtain the required location information.

Obviously, the strategies mentioned above mainly target on some specific situations. For example, the forwarding strategy focuses on the situation where the rate of mobility is higher than the rate of call arrival. In order to further reduce the loading of a centralized root database like HLR, the most effective mechanism is to introduce

multilevel database architecture. However, the increase in levels will result in longer delays in location registration and call delivery. In view of such tradeoff, a three-level multitree database architecture has been proposed [233].

Figure 24.9 illustrates the proposed multitree distributed database architecture. In the figure, there are a number of distributed database subsystems (DS). Different DS are probably managed by different service operators. DS can communicate with other DS through the PSTN network, ATM network or other networks with the help of the root database DB0 of their own DS. In this architecture, DB0 and DB2 correspond to the HLR and VLR in the GSM network, respectively. Similar to the GSM network, DB2 is located together with the MSC, which controls an RA (registration area). As a result, when a mobile terminal moves from one RA to another within the same MSC, location update can be avoided. A number of DB2s are grouped together by DB1, which contains a switch called STP to route messages between various databases. Similarly, a number of DB1s are grouped together by the root database DB0. DB0 contains the entries of all the users in the GSM network and service profiles of users within the service area. Each entry in DB0 contains a pointer to either other DS' DB0 when a user is within other DS or to current DS' DB1 when the user is within the same service area. DB1 then contains an entry which has a pointer to DB2 where the user is currently visiting or roaming. The corresponding DB2 has a copy of the service profile of the user.

With the use of the above-mentioned architecture, the loading of the root database can be reduced. This will be explained in the following location update and call delivery procedures. First of all, let us discuss the location update procedures:

1. When a mobile terminal moves from one RA to another RA managed by a different DB2, the registration message is sent to the new DB2.

2. The new DB2 figures out that the user enters the RA from other RA managed by other DB2, it then forwards the registration message to DB1.

3. In DB1, if it can find an entry in its database, it simply updates the pointer to the new DB2 and sends a deregistration message to the old DB2.

4. However, if an entry cannot be found in DB1, then it knows that the user is originally managed by a different DB1. In this case, a new entry is added in the new DB1. The new entry created points to the new DB2. Then a registration message is forwarded to DB0.

5. If an entry is found in DB0, then the user's movement is within the same DS. In this case, DB0 updates the pointer of the user to the new DB1 and then sends a deregistration message to the old DB1. The old DB1 then sends a deregistration message to old DB2 and deletes the entry in its database.

6. However, if an entry is not found in DB0, the user has crossed DS. In this case, the new DB0 requests for the user profile of the user from the old DB0. The

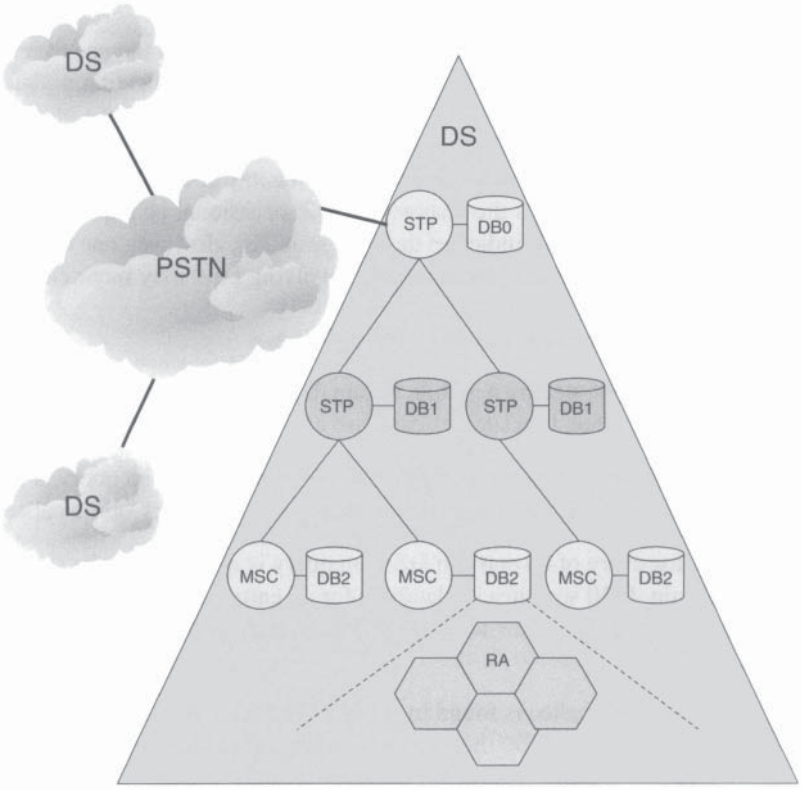


Figure 24.9 Multitree distributed database architecture.

new DB0 then saves the user service profile in its own database and adds an entry to the new DB1, which then points to the new DB2 where the user is present.

7. In the above procedures, if the user's movement is within the same DS, then the copy of a user service profile can be directly sent from one DB2 to another DB2. However, if the movement crosses DS, the service profile of the user is required from the new DB0. After the deployment of the service profile for the user, the registration process is regarded as completed.

The above steps describe the location update procedures with the use of a three-level multitree architecture. With the help of this architecture, location updates on the root database are greatly reduced. The main reason is that the use of multiple levels allows updates to be done in the intermediate level database instead of sending update messages to the root database. In particular, the movement of a user from one RA to another usually involves the change of DB2, occasional change of DB1 and infrequent change of DB0 (this indicates the change of DS). However, one of the main drawbacks of this architecture is that a change involving DB0 may incur changes in DB0 of all DS.

The call delivery procedures are described as follows:

1. When a call is initiated by a mobile terminal and received at caller's MSC, the collocated DB2 then looks for its database entry for the corresponding callee.

2. Without discovery of an entry in DB2 of the caller's MSC, an query is sent to the DB1, and the DB1 also searches its database for an entry of the callee.

3. Without discovery of an entry in DB1, an query is again sent to its associated DB0. Again, DB0 searches its database for an entry of the callee. The search will be repeated in callee's located DB0 if an entry is still cannot be found in caller's DB0.

4. If the entry of the callee is found in caller's DB0 (i.e., the caller and callee are within the same DS), DB0 then sends a routing address request to the callee's associated DB1, which then sends the request to the associated DB2.

5. On discovering the entry of the callee in DB2, a TLDN is assigned to the callee as the routing address and the TLDN is transmitted back to the MSC of the caller. After receiving the TLDN, the caller's MSC can then find the callee and its associated MSC to set up the call.

Similar to that in the location update procedures, intermediate databases can be used to help reduce the loading of the root database whenever the caller and callee are within the same DS.

With the help of the distributed database subsystems architecture discussed above, the signalling load is much reduced. This leads to the possibility to support a larger

amount of users. As a result, the system becomes more robust and scalable when compared with the original design of GSM HLR-VLR architecture.

24.8 SUMMARY

In this chapter, the design principles of improving the availability of files in a distributed file system have been discussed. Afterwards, the Coda file system has been discussed in detail. We have discussed the server replication method with use of optimistic replication in Coda. Then, the ideas of disconnected operations and weakly connected operations with the use of hoarding, emulation and reintegration are discussed in detail.

The success of the Coda file system indicates that two very different techniques in improving the availability of files—replication and caching, can actually be combined to cooperatively improve both the quality and availability of files. This is particularly important to the users of mobile portable devices like laptops, PDAs, etc., which may only get limited network access.

Finally, as practical illustrations, the idea of using mobile distributed databases in a distributed multitree database architecture in replacement of the generally adopted GSM HLR-VLR architecture has been discussed in detail. It is found that the use of the multitree architecture can effectively reduce the signalling load when comparing to the conventional GSM HLR-VLR architecture.

PROBLEMS

24.1 Briefly discuss the general concept of a distributed file system.

24.2 What is optimistic replication? How is it different from pessimistic replication?

24.3 Justify the use of server replication and disconnected operations in a distributed file system.

24.4 What are the differences between disconnected operations and weakly connected operations?

24.5 How can a distributed multitree database architecture be used in a cellular network with location updates? Briefly discuss why it can obtain better performance than the GSM HLR-VLR architecture.

This Page Intentionally Left Blank

CHAPTER 25

LOCATION DEPENDENT SERVICES

The ideal of service is the basis of all worthy enterprise.

—Principles of Rotary, I, 1905

25.1 INTRODUCTION

Anytime, anywhere wireless access does not only enable uninterrupted connectivity to users, but also the possibility of continuous location tracking. Naturally, service providers or other organizations (e.g., government, law enforcement, rescue operations, etc.) would like to exploit such location information so as to provide additional services.

Interoperability:

There are two main types of mobile location services—in-vehicle mobile location service and personal mobile location service. Although their development objectives are different, they require similar components in supporting their service. The components include map database service, spatial analysis service and mobile positioning

Wireless Internet and Mobile Computing. By Yu-Kwong Ricky Kwok and Vincent K. N. Lau

Copyright © 2007 John Wiley & Sons, Inc.

service. The interoperation of these components contributes to the correct calculation of the location of the user, which is discussed in Section 25.2. Specifically, the spatial analysis service is used for the translation of map information to number and vice versa, while the mobile positioning service is used to estimate the user's current location. The techniques of these two services are discussed in Section 25.3 and Section 25.4, respectively.

Performance:

There are a number of mobile location services envisioned by service providers. These services require different requirements on the accuracy of user's location estimation, response time, and the need for in-door estimation. These requirements are, to some extend, governed by the kind of mobile positioning techniques adopted. This is discussed in detail in Section 25.4.

25.2 MOBILE LOCATION SERVICE MODEL

Generally, mobile location services can be classified into two main types—in-vehicle and personal mobile location services, as illustrated in Figure 25.1 [166].

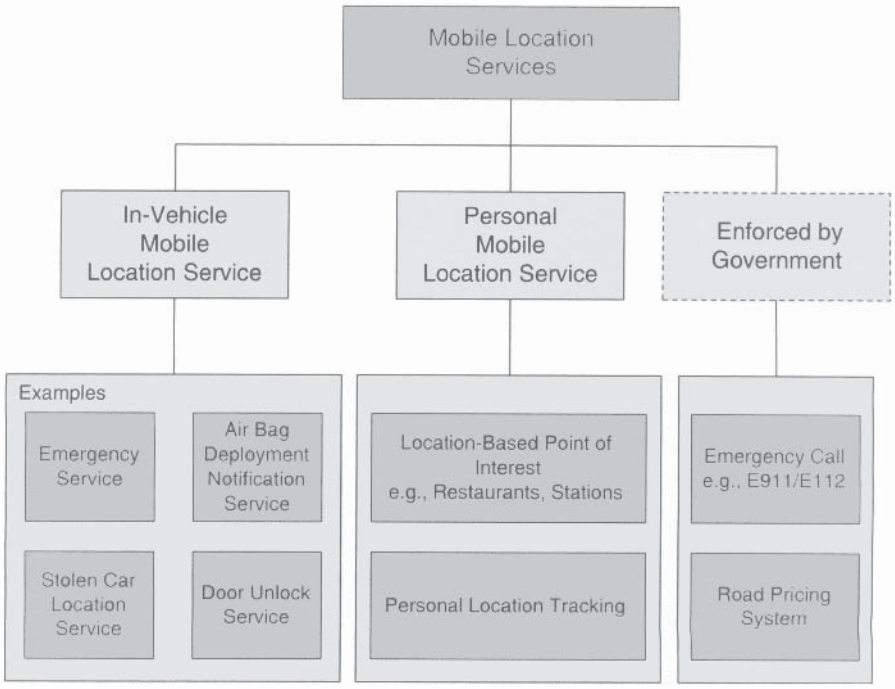


Figure 25.1 Mobile location service model.

25.2.1 In-Vehicle Mobile Location Service

The in-vehicle mobile location service refers to the location service specially designed for automobiles [166]. This service is mainly driven by automobile manufacturers who want to maintain a close relationship with their customers and to generate more revenue by providing services to their customers. For example, in some interactive services, assistance can be provided to drivers when some emergency events occur.

Some of the services provided by OnStar [263], which is a company providing mobile location services for automobiles, are examples of in-vehicle mobile location services:

Emergency Service: In this service, the driver can ask for assistance with the system in the vehicle. Upon receipt of the call, if any further assistance is needed (e.g., ambulance, police, etc.), the service assistant can help by referring the case to the the corresponding emergency department on behalf of the driver.

Air Bag Deployment Notification Service: In this service, the service assistant will contact the driver directly when the system detects that the air bag of the vehicle is deployed. If there is on response from the driver, the emergency department will be contacted.

Stolen Car Location Service: Through this service, the owner of the vehicle can track the location of his vehicle when it is stolen.

Door Unlock Service: When the owner of the vehicle leaves his key inside the vehicle with it locked, he can ask for assistance from the service provider through a phone call.

In fact, besides providing assistance to the driver/owner at the right time, in-vehicle mobile location service also includes some entertainment, Internet access service, weather report service, etc. However, the main focus is to provide a safe driving experience to the customers.

25.2.2 Personal Mobile Location Service

Personal mobile location service is designed for persons, rather than vehicles [166]. The aim is to provide a convenience service to a user located in a particular area. For example, when the user goes to a certain area that he is not familiar with, the map of the area is particularly useful to the user. In addition, a service like personal location tracking and verification is also useful for some emergency scenarios. These services are mainly provided by mobile operators through common communication devices like mobile phones, PDAs installed with GSM cards, etc. Furthermore, these services can be provided with the existing mobile infrastructure coupled with some add-ons. We will discuss this in more detail in Section 25.4.

The following is one of the practical examples of personal mobile location service provided by NTT DoCoMo [89]: When the user arrives at a certain area, through his

mobile device he can obtain information like map of the area, locations of restaurants, local weather report, town information, etc.

25.2.3 Government Enforcement

In addition to in-vehicle and personal mobile location services provided by automobile manufacturers and mobile operators for commercial purposes, there are some mobile location services enforced by the government. For example, E911/E112, and road pricing. E911/E112 services[1] are the emergency services suggested by the government such that mobile operators need to handle local 911/112 calls no matter whether the calls are initiated from the mobile phone subscribed to their service or not. On the other hand, road pricing is suggested in some countries in which vehicles are required to be equipped with some location tracking devices. In his way, the road users can be charged with appropriate fares.

25.2.4 Components of Mobile Location Services

In order to support the mobile location services that we have discussed in previous sections, the components as illustrated in Figure 25.2 are required. First of all, because mobile location services require location information of users up to certain degree of accuracy, a location service is necessary. The location service includes three components—map database service, spatial analysis service and mobile positioning service [166]. Map database service is necessary whenever the location of the user is required to be figured out. On the other hand, spatial analysis service is required to map the position represented some keywords to some coding and vice versa. In addition to this, the exact location of the user can be found out by the mobile positioning service. In subsequent sections, we will discuss in detail the spatial analysis and mobile positioning techniques.

25.3 SPATIAL ANALYSIS TECHNIQUES

The spatial analysis service comprises several components, including map database, geocoding, reverse geocoding, routing, map formation, point of interest, map attributes modification, etc.

As Figure 25.3 [166] illustrates, the map database is involved in various operations of the spatial analysis service. There are two types of map model used in the map database for storage of data—entity data model and continuous variation data model. In the entity data model, points, lines and polygons are used for representing real-world objects or entities with some associated attributes predefined. For example, specific locations like a subway station can be represented by points, a road or street

[1]E112 services are optional in some European countries.

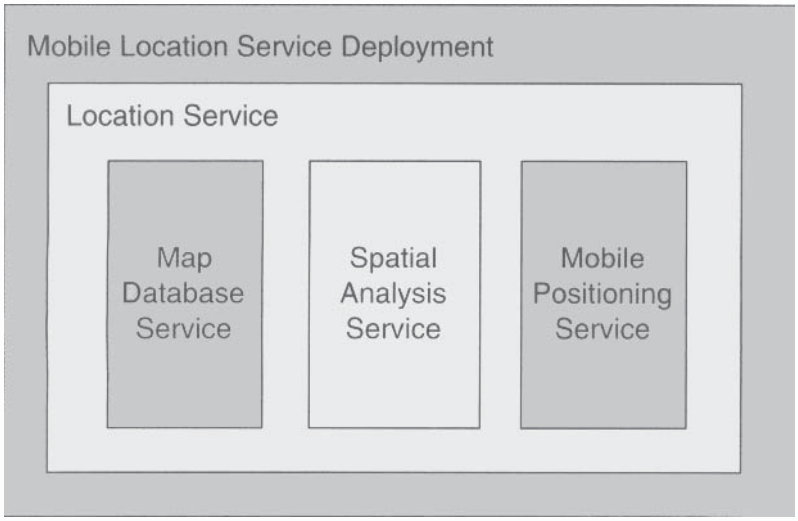


Figure 25.2 Mobile location service components.

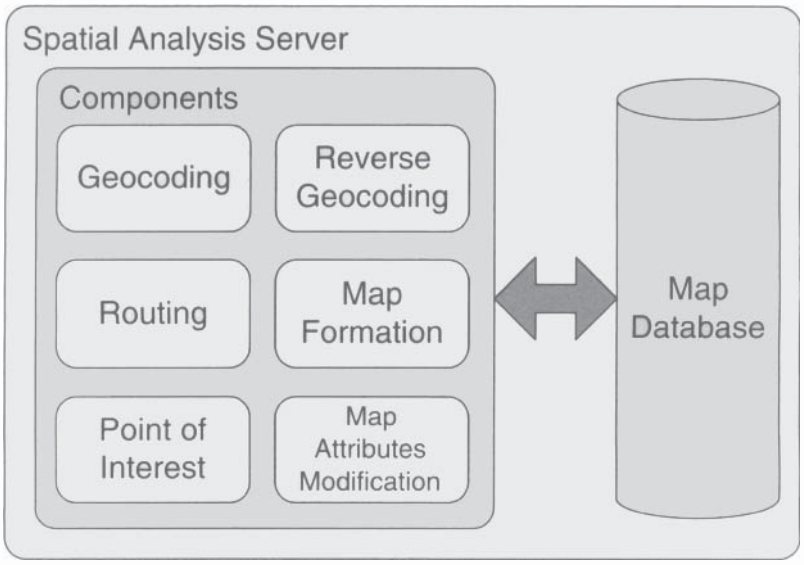


Figure 25.3 Spatial analysis service components [166].