COMPSCI 314 S1 C

Wireless LANs
Introduction to IP and Routing

314s1c Terms Test

- 6.25 pm, Monday 8 May 06 (next Monday)
- Test rooms
  - PLT1 (303-G20) Surnames A - L
  - MLT1 (303-G23) Surnames M - Z
- Format of test
  - One-hour test, short answer questions
  - Test covers material presented in lectures for *first half of semester* (i.e. this week's lectures are *not* included)
  - 2005 paper (with model answers) is on 'tests and assignments' course page
- *Tutorial: Monday 8 May (test day, instead of lecture)*

Wireless Networks

- We only look at Halsall chapter 4, sections
  - 4.1 Intro: network types (PAN,LAN, cellular)
  - 4.3 802.11 networks
  - 4.4.1 GSM network overview
- Essential differences from wired networks
  - Wired fast and reliable
  - Wireless slower and less reliable
- Interaction between wired and wireless
  - 802.11 exchanges packets directly with 802.3 (Ethernet)
  - GSM 'phones communicate directly with POTS 'phones

Wireless Personal Area Network (WPAN)

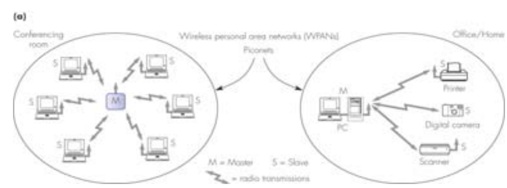


Figure 4.1 Wireless networks: (a) piconets/wireless PANs

Wireless LAN

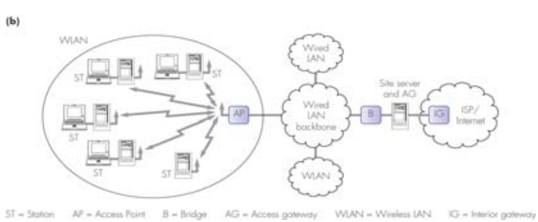


Figure 4.1 Wireless networks: (b) wireless LAN

Cellular network topologies

Figure 4.1 Wireless networks: (c) cellular/mobile radio networks

Wireless LAN *(managed mode)*

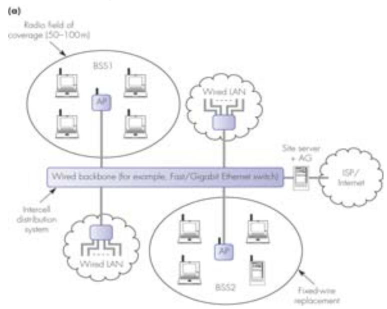


Figure 4.11 IEEE802.11 operational modes: (a) infrastructure

Wireless LAN *(ad hoc mode)*

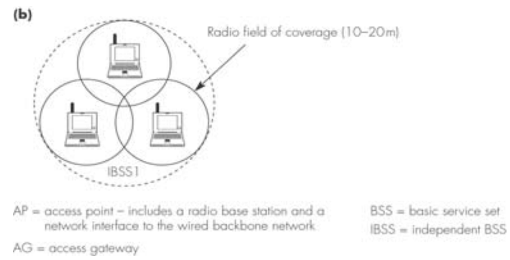


Figure 4.11 IEEE802.11 operational modes: (b) ad-hoc

802.11

- IEEE standard
- Most common wireless system for laptops
- Uses same data frame format as Ethernet (802.3)
  - Allows wired and wireless LAN segments to be intermixed
  - Ethernet packets are simply passed between the wired and wireless MAC layers
- MAC layers differ
  - Ethernet (802.3) hosts can reliably detect collisions. If a packet doesn't collide, it is delivered to its destination
  - Packets sent by wireless (802.11) hosts may not arrive, and/or collisions may not be sensed reliably
  - 802.11 uses 'control' packets to manage the (wireless) medium

802.11 Standards

- 802.11b
  - Most common, 11 Mb/s, HR-DSSS encoding
- 802.11a
  - Various speeds up to 54 Mb/s, OFDM, 52 carrier frequencies
- 802.11g
  - 54 Mb/s, an improvement on 802.11a

- Most 802.11 implementations (e.g. PC wireless cards) are *backwards compatible,* i.e. can support the earlier standards

802.11 MAC layer, Distributed Coordination Function (DCF)

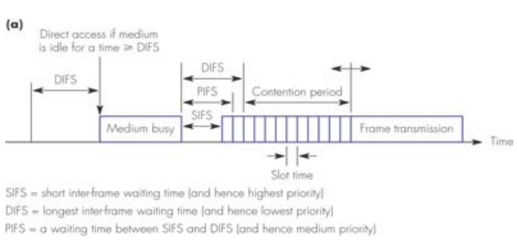


Figure 4.13 Operation of the basic DCF with CSMA/CA in the broadcast mode: (a) definition of the timing parameters

802.11 MAC (2)

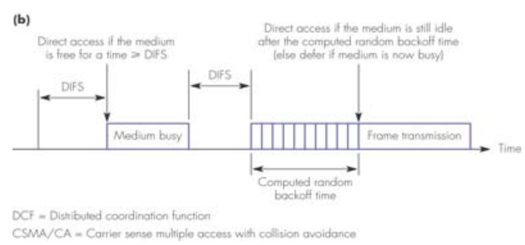


Figure 4.13 Operation of the basic DCF with CSMA/CA in the broadcast mode: (b) use of random backoff

802.11 MAC (3)

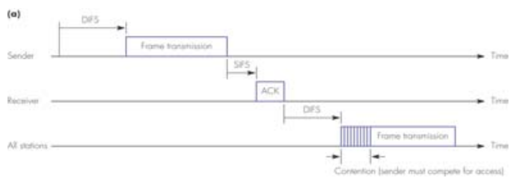


Figure 4.14 Operation of basic DCF with CSMA/CA in the unicast mode: (a) successful frame transmission

802.11 MAC (4)

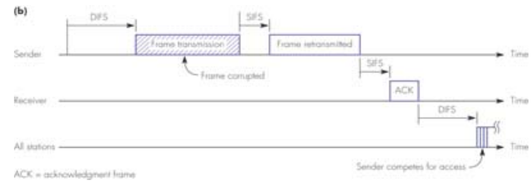


Figure 4.14 Operation of basic DCF with CSMA/CA in the unicast mode: (b) retransmission procedure

802.11 Hidden Station

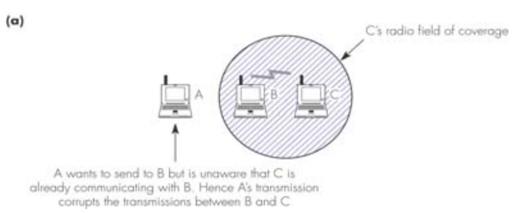


Figure 4.15 DCF with RTS/CTS extension: (a) hidden station problem

802.11 Hidden Station (2)

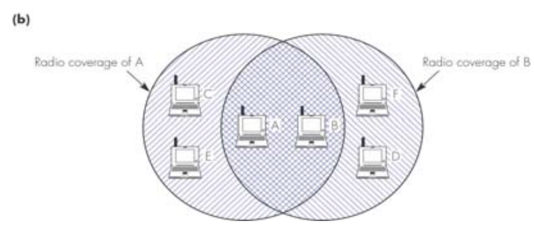


Figure 4.15 DCF with RTS/CTS extension: (b) example network configuration

802.11 Hidden Station: timing

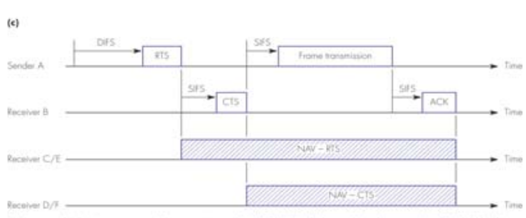


Figure 4.15 DCF with RTS/CTS extension: (c) associated timing diagram

Wireless Frame Fragmentation

- Because the wireless medium is less reliable, 802.11 breaks *Ethernet frames* into smaller *802.11 segments*
- Using smaller segments
  - Improves the probability they will arrive without errors
  - Reduces the amount of data to retransmit after detecting an error
- We're not going to look at the details

802.11 MAC Frame Formats

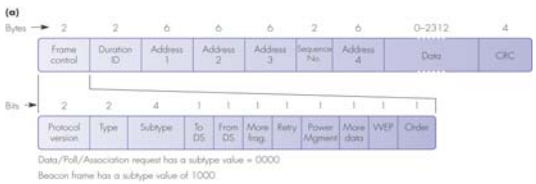


Figure 4.18 MAC frame formats: (a) data and management frames

802.11 MAC Frame Formats (2)

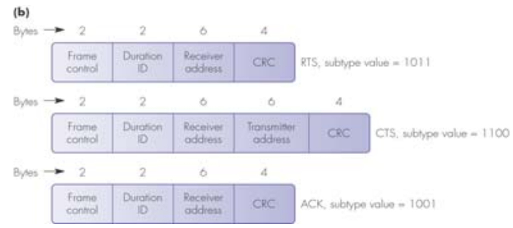


Figure 4.18 MAC frame formats: (b) control frames

Infrastructure Mode: Moving Hosts

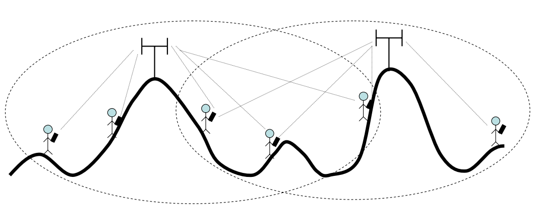
- A mobile host (e.g. laptop with wireless card) can move from one access point (AP) to another
- Need to 'hand over' a moving station from old AP to new AP
  - To reach a particular mobile host (MH), network always needs to know which AP is looking after it
  - Old AP must tell new AP it is now responsible for MH
  - It must also tell other network switches to send MH packets to new AP

Cellular Networks: GSM Overview

- Halsall section 4.4.1
- We're not going to look into the details of GSM
- Figure 4.20 shows the architecture of a (2G) GSM network
- Instead of using that figure, here are some slides originally prepared by Ulrich Speidel …

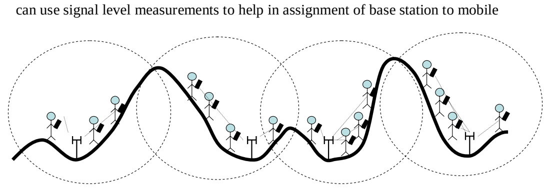
GSM Cell Siting

- Early phase of network development – few users, few sites, little pressure on frequency resource, large cells
- Cell base stations are mainly sited on hilltops, large buildings in order to maximise coverage per cell

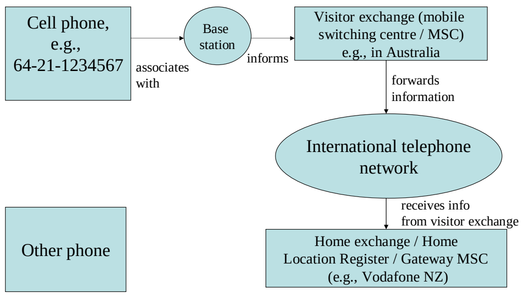


GSM Cell Siting (2)

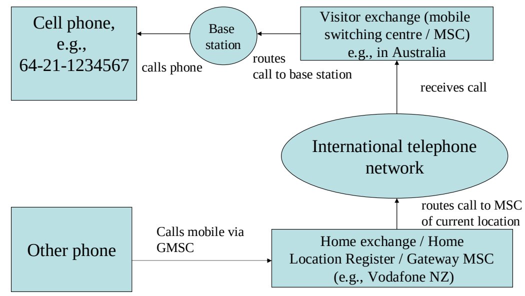
- Mature phase of network development – many users, many sites, lots of pressure on frequency resource, small cells
- Cell base stations are mostly situated at ground level at the bottom of valleys – use hills or buildings as shields to limit coverage/number of users in cell and enable frequency re-use on other side of hill
- 'Umbrella cells' on hilltops can provide coverage for areas not covered by other cells – can use signal level measurements to help in assignment of base station to mobile



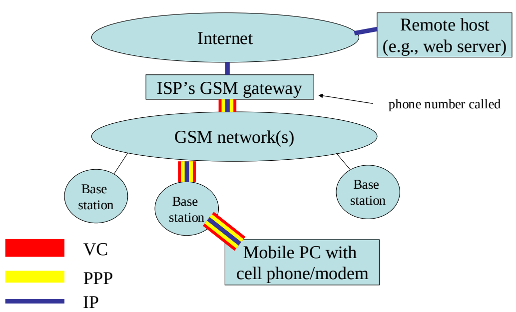
Keeping Track: Mobile Node informs MSC



Keeping Track: Incoming call to Mobile Node



Mobile Computing via GSM Cell Phones



IP, the Internet Protocol

- Halsall sections 6.1 – 6.4
- Defined in 1981,RFC791 (available from http://www.ietf.org/)
- *Version 4* is now deployed as the predominant Internet protocol – unless otherwise mentioned, this is the version we discuss here
- Part of a suite of protocols that has collectively become known as TCP/IP
- Provides *connectionless best-effort* packet delivery between *hosts* on the Internet
- Hosts have unique addresses, in blocks allocated by ICANN (Internet Corporation for Assigned Names and Numbers)

IP: Overview

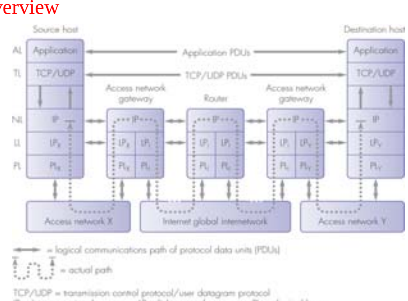


Figure 6.1 Internet networking components and protocols

IP: the Protocol Family

Note: only 5 layers in the IP stack

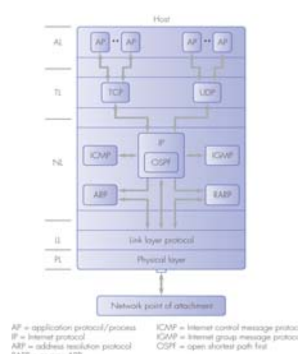


Figure 6.2 IP adjunct protocols

IP addresses [section 6.4]

- An IP address identifies an *interface* on a host (i.e. on a network node)
- One host can have multiple IP addresses, but one IP address cannot be assigned to more than one interface on one host
- An IP address has two parts: *network* and *host*. A network mask (*netmask*) indicates the network bits
- IP addresses originally came in five flavours: Class A, … E
  - First few bits indicate which class an address belongs to
- CIDR (Classless Inter-Domain Routing) makes those ‘classes’ obsolete
  - Address and address length are two parts of a *netid* (CIDR block, or prefix)
  - Routing protocols must carry both parts of every netid

IP addresses (2)

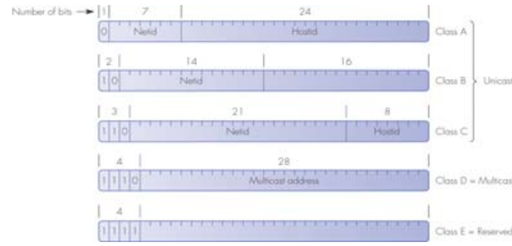


Figure 6.5 IP address formats

Class-based IP Addresses

Class A: First octet designates network, last three octets designate host within the network (netmask 255.0.0.0). Network contains 2^{24} addresses. First bit in first octet of address is always zero

Class B: First two octets designate network, last two octets designate host within the network (netmask 255.255.0.0). Contains 65536 addresses. First octect starts with 10

Class C: First three octets designate network, last octet designates host within the network (netmask 255.255.255.0). Contains 256 addresses. First octet starts with 110

Class D: Multicast address (packet gets sent to more than one host). First octet starts with 1110

Class E: Reserved for future use (unlikely). First octet starts with 11110

CIDR addresses

CIDR (‘cider’) = Classless InterDomain Routing

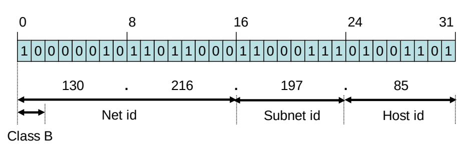
- A newer concept where any number of leading bits can designate the network, with the remaining bits designating the host
- Written with a trailing slash indicating the number of network bits
- e.g.: 130.247.156.87/22 is a host on a network with 2^{32-22} = 1024 addresses

Another example

The University of Auckland has a Class B address, 130.216.0.0/16

We run the network as though it were a set of Class C *subnets,* 130.216.0.0/24

Network addresses like these (with trailing zeros) are usually called *network prefixes* or *netids*



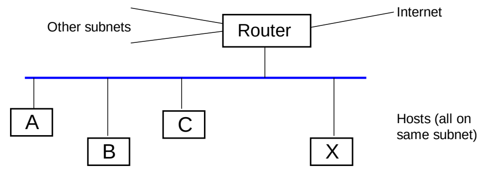
NAT: Network Address Translation [Section 6.4.4]

- Maps IP addresses inside a network to other addresses outside it
- Commonly used as a way to map IP address/port pairs so that many ‘inside’ hosts can use a much smaller number of ‘outside’ addresses
- Various views of NAT
  - An effective way to extend IPv4 address space
  - Hides ‘inside’ network structure
- But ..
  - Breaks the ‘end-to-end’ principle, i.e. protocols and applications may need to know about NAT boxes in the host-to-host path
- Halsall says it’s just another way to use addresses

Special IP addresses

- Network or host bits all set to zero – refers to own host or network
- 255.255.255.255 (all bits set to 1) – broadcast address.
  Used, e.g., in BOOTP and DHCP for host configuration
- 127.0.0.1 – local loopback address referring to the host itself
- 10.0.0.0 - 10.255.255.255 (10/8 prefix),
  172.16.0.0 - 172.31.255.255 (172.16/12 prefix),
  192.168.0.0 - 192.168.255.255 (192.168/16 prefix)
  These are private addresses reserved for private Internets. Defined in RFC 1918. Very commonly used in conjunction with Network Address Translators (NATs) behind enterprise gateways.
- Firewalls (filtering gateways) are usually configured to drop packets to these addresses as they are not supposed to be used outside the local network

IP host configuration



- Each host knows its *address* and *netmask* or *netid*
- It also knows the IP address of its default router
- As well, it will need to know the IP address of a *nameserver* (more later)

Mapping IP addresses to MAC addresses [Section 6.6.2]

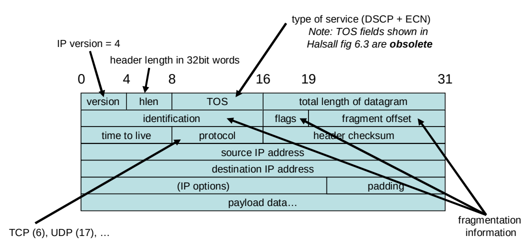
- How do hosts on a shared medium,
  e.g. Ethernet, recognize that an IP datagram is for them?
- First approach: Could *always* broadcast to all hosts on the network; each host decodes the packet and looks at the IP address: Lots of decoding overhead and what do we do with bridges?
- Second approach: Find out the MAC address of the host first. Need some way of doing this: The Address Resolution Protocol (ARP)

Address Resolution Protocol (ARP): 6.6.2

- Part of the 'TCP/IP family,' Defined in RFC 826
- Gateway or other host that wants to find out a MAC address for an IP number broadcasts an ARP request throughout the (local) network. This broadcast is received by all hosts in the network
- Each host hands this packet to its ARP implementation
- If the IP address in the packet does NOT match the IP address of the host, the host remains silent
- If the IP address matches that of the host, the host replies with an ARP response packet that contains both the IP address and the MAC address of the host
- The requesting host can now cache this mapping for future use

IP packet header [Section 6.2]

- The IP packet header bytes precede the payload data in the IP datagram (= IP packet)

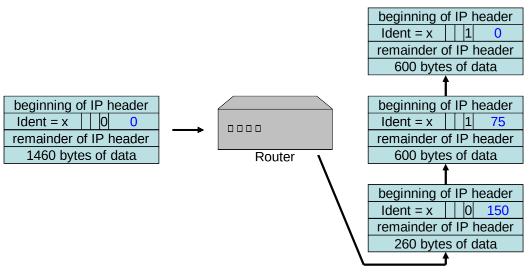


IP fragmentation and reassembly [Section 6.3]

- Any router along the path may split an IPv4 datagram into two or more fragments
- Each fragment becomes its own IP datagram with its own header
- All fragments are given a common *ident* field – in fact the same ident as the original datagram
- Third bit ('M' bit as in 'more to follow') in the flags field is set in all but the last fragment
- The fragment offset field contains the offset of the fragment's payload within the original payload,
  in units of 8 bytes.
- All fragments travel separately to the final destination (i.e., they do not get reassembled by the next router with larger MTU)

Example: IP fragmentation

Example: MTU of onward link frame only permits 600 bytes of payload after IP header in frame



Another fragmentation example

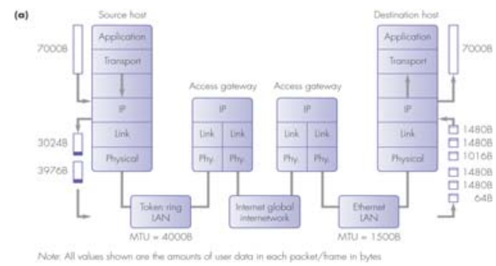


Figure 6.4 Fragmentation and reassembly example: (a) Internet schematic

Another example (2)

(b) *Token ring LAN:*

	(i)	(ii)
Identification	20	20
Total length	7000	7000
Fragment offset	0	497
(User data)	3976	3024
M-bit	1	0

(c) *Ethernet LAN:*

	(i)	(ii)	(iii)	(iv)	(v)	(vi)
Identification	20	20	20	20	20	20
Total length	7000	7000	7000	7000	7000	7000
Fragment offset	0	185	370	497	682	867
(User data)	1480	1480	1016	1480	1480	64
M-bit	1	1	1	1	1	0

Figure 6.4 Fragmentation and reassembly example: (b) packet header fields for token ring LAN; (c) Ethernet LAN

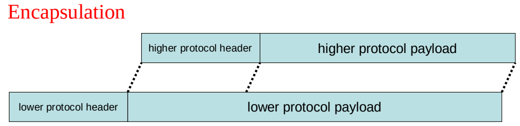
Reassembly after fragmentation

- Reassembly **always** happens at the final destination, even if MTU size increases again
- Fragments may have taken different paths and may arrive in different order!
- Further fragmentation of already fragmented datagrams is possible – in this case, if the M bit is already 1, the M bits in the flags of all additional fragments are set to 1
- May need to buffer fragments for a while until the missing parts arrive

Shortcomings of IP

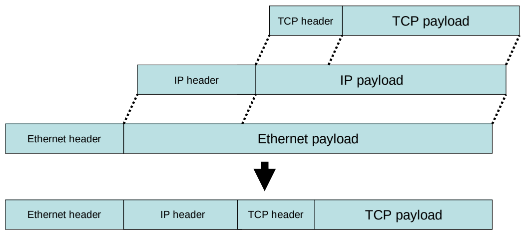
- Severely restricted address space in IPv4 (solved in IPv6)
- IP address reflects physical network topology. If a node moves from one part of the network to another, the IP address cannot stay the same. Bad news for mobile routing.
  (Current work on *Host Identity Protocol* could solve this)

Encapsulation



- It is often necessary to transfer data across different network formats
- For example, IP over ATM, TCP over IP, IP over Ethernet, IPv4 over IPv6, IPv6 over IPv4, etc.
- Encapsulation puts one protocol's packet into the payload field(s) of another protocol's packet(s)
- Very widely practised – get used to the idea!
- Halsall has no section on encapsulation – he seems to regard it simply as an implementation detail of the networking stack

Example: TCP over IP over Ethernet



… anyway, we have yet to look at the details of TCP