COMPSCI 314 S1 C Assignment 4

Department of Computer Science
The University of Auckland
Due Wednesday 24 May 06, 11:59 pm

This assignment will contribute 4% to your overall course mark.

Submit your assignment via the DropBox, either in PDF (preferred), or in MS Word format.

These are all short-answer *questions, one or two sentences (or a short calculation with some comments) are all that is needed to answer them. Read each question carefully right through* before *you begin to answer it!*

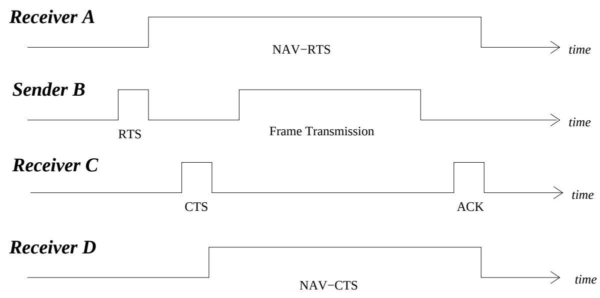
1. **Wireless LANs** [13 marks]

   In an 802.11 network we assume that all the stations are similar – they have the same transmit power and receive sensitivity, so they can all work over the same maximum distance between stations.

   (a) Briefly explain what is meant by the *hidden station* problem. [2 marks]

   > Assume that four stations are arranged more or less in a straight line, in order A–B–C–D. C can receive from B, but not from A, so A and C are *hidden* from each other. Similarly, B and D are hidden from each other.
   > One mark for explaining *hidden station*.
   > Another mark for using an example like this, with *two* pairs of hidden stations.

(b) Explain, with the aid of a timing diagram, how 802.11's RTS/CTS extension solves the hidden station problem. [4 marks]



This diagram (a modified version of *Halsall's* figure 4.15c) shows a timing diagram for 802.11 RTS/CTS. It assumes the same arrangement of stations as in part (a) above:

- B sends RTS, receives CTS, then sends its frame.
- When A has received B's RTS, it knows it can't send until B has finished. It sets the time it must wait, NAV from the value given in the RTS frame.
- C receives CTS, receives B's frame, then sends ACK.
- D receives C's CTS, and knows it can't send until C has sent its ACK. It sets its own NAV value, i.e. the time it must wait, from the value given in the CTS frame.

Note that A and D must set their NAVs then wait that long; they can't rely on hearing C's ACK, or even B's frame transmission.
Marks: one for the diagram, two for explaining RTS/CTS, one for saying that NAVs are needed to recover from timeouts.
(zero marks if you cut/paste a diagram without making proper reference to its source).

(c) Consider a situation where stations B and C are near each other; *station A is close to B, but out of range for C and D,* and *station D is close to C, but out of range for A and B.* Is it possible for C to transmit to D at the same time as B is transmitting to A? Explain how this can be achieved. [3 marks]

- Assume that B sends an RTS to A, i.e. specifying A in the Receiver Address field of an RTS frame. C will hear the RTS, and know that B is sending to A. C will not respond, but it has a NAV from the RTS, so it knows how long it will be before it could try sending to B.
- Now C could send an RTS to D. If it did, D could reply with a CTS, allowing C to transmit its data. However, while C was sending, B wouldn't hear it because it's busy sending to A. When B finished sending, it might not be able to hear A's ACK, because C's transmission could interfere with it.
- Summary: it's possible in principle, but the RTS/CTS extension doesn't support it. You could allow it by turning RTC/CTS off, but that could reduce efficiency because of collisions and backoffs from them.

Marks: One for each of the three points above.

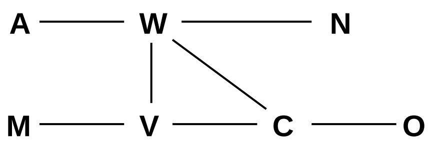
(d) In 802.11, RTS/CTS capability is an optional extension. Is it possible to operate a wireless network with RTS/CTS turned *off?* When might it be worthwhile to do that? What advantage might be gained by doing so? [4 marks]

- Yes, at least in principle.
- You could do this when you know that every station in your network can hear transmissions from every other station, e.g. a group of laptops in the same room as their base station.
- Turning it off would mean stations only have to wait until no other station is transmitting, before they can start to transmit (i.e. the network can use only CSMA/CA protocol). Since they don't have to send RTS/CTS, network overhead is reduced, so users might get slightly better performance (improved latency).
- In practice there's very little performance gain, so everyone simply leaves RTS/CTS on.

Marks: One for each of the four points above.

2. **Distance Vector Routing** [14 marks]

The following diagram shows the New Zealand Universities' network as it was in 1998. The letters indicate sites, e.g. 'A' is the U Auckland network, 'C' is the Canterbury network, etc., and the lines represent 48 kb/s links between the sites.



(a) Assume that the network is initialised by powering on all of its routers at the same time – i.e. all routing tables start from their initial state – and that routes are exchanged using RIP at $30s$ intervals.

Make a set of routing tables showing, for each site, a list of the other known sites, their distance (in hops), and the link used to reach them (e.g. by specifying the 'next-hop' site). *Use a spreadsheet, and/or a fixed-width font to make the tables.*

This set of tables should show the network in it's initial state, i.e. before any routing information has been exchanged. [3 marks]

Matrix is set out as in the lecture slides, for each site (pair of columns) I show next-hop site and number of hops to (destination) site.

	A	W	N	M	V	C	O
A	-	W 1	?	?	?	?	?
W	A 1	-	N 1	?	V 1	C 1	?
N	?	W 1	-	?	?	?	?
M	?	?	?	-	V 1	?	?
V	?	W 1	?	M 1	-	C 1	?
C	?	W 1	?	?	V 1	-	O 1
O	?	?	?	?	?	C 1	-

Marks:

- All hop counts are 1
- A,N, M,M rows have only one entry (others are ?)
- W, V and C have 4,3 and 3 entries

Deduct one mark if any entry is the matrix is incorrect.

(b) After 30s, all sites exchange routing information. Show their routing tables after this first round is complete. [3 marks]

	A		W		N		M		V		C		O	
A	-		W	1	W	2	?		W	2	W	2	?	
W	A	1	-		N	1	V	2	V	1	C	1	C	2
N	W	2	W	1	-		?		W	2	W	2	?	
M	?		V	2	?		-		V	1	V	2	?	
V	W	2	W	1	W	2	M	1	-		C	1	C	2
C	W	2	W	1	W	2	V	2	V	1	-		O	1
O	?		C	2	?		?		C	2	C	1	-	

Marks:

- Changed entries (shown in red) *all* have hop count 2
- A,N, M,M rows have only one next-hop site
- Only the O and M rows and columns have ? entries remaining

Deduct one mark if any entry is the matrix is incorrect.

(c) How will the routing tables change after the third round? [2 marks]

After the second round the matrix looks like this . . .

	A		W		N		M		V		C		O	
A	-		W	1	W	2	W	3	W	2	W	2	W	3
W	A	1	-		N	1	V	2	V	1	C	1	C	2
N	W	2	W	1	-		W	3	W	2	W	2	W	3
M	V	3	V	2	V	3	-		V	1	V	2	V	3
V	W	2	W	1	W	2	M	1	-		C	1	C	2
C	W	2	W	1	W	2	V	2	V	1	-		O	1
O	C	3	C	2	C	3	C	3	C	2	C	1	-	

After that all sites/routers have a complete routing table, so nothing will change until the network topology changes, e.g. because a link or a router fails.
Marks: one for saying that routing tables won't change after third round, one for a brief explanation of *why* they won't change.

(d) What routing changes will happen if the link between sites 'W' and 'V' fails? How many routing exchange rounds will it take before the routing tables are stable again? [4 marks]

- W and V sites see the link failure, and take destinations that used W or V as next-hop out of their routing tables. (1 mark)
- The stub (singly-connected) sites, i.e. A,N,M and O, never change any of their next-hops. In one round, W and C exchange their routes, and all sites have the correct next-hop for all the other sites. (2 marks)
- It takes a second round for all sites to get their hop counts correct, after that the tables are stable. (1 mark)

(e) Assume that before the outage described above, a large file transfer was taking place between sites 'A' and 'M.' What would happen to the packets being sent during the outage? [2 marks]

It depends on what transport protocol was sending the packets.

- If it was UDP, the routers may hold the packets in their output queues, to be sent when the link is restored. If the hosts continue to send during the routing round, router buffers may overflow, causing packets to be lost.
- If it was TCP (ensuring reliable delivery), the sender will detect packet losses, slow down to a complete stop (when sender transmit buffers are full, wait for routing (packet transport) to be restored, then resend lost packets. Apart from the delay during the routing rounds, the user will not notice the outage.

1 marks for either of the above answers, 2 marks for both.

3. **IP Addressing (IPv4 and IPv6** [13 marks]

(a) Describe briefly an IPv4 address. You should assume that the address is given in CIDR format, and explain which bits are used to identify a network, a subnet and a host. [2 marks]

- First n bits are the *network number*
- Remaining bits are the *host number*
- Hosts have to know what n is for the network they're connected to

- Network administrator may choose to use the first s bits of host addresses to identify a host's *subnet*
- Hosts within the subnet use the first $n + s$ bits as their network number

Marks: one for each of the two lists of points above.

(b) Suggest a strategy for allocating addresses within an IPv4 subnet; assume you are using network 192.168.0.0/16, and allocating space within one of its 20-bit subnets. [2 marks]

- We have a 16-bit network id, and a 20-bit subnet width. That gives us 16 subnets, 0..16. It's common (though not neccessary) not to use the highest and lowest subnets, which leaves subnets 1..14.
- Choose an address for the subnet's *default router*, e.g. the highest possible address. For subnet 1 that would be 192.168.31.254 (note, we can't use 31.255, that would be the subnet broadcast address).
- Allocate host addresses starting at 1, and increasing from there. Or, perhaps, allocate addresses sequentially in groups, say one for each room in a building.
- We'd write host addresses in subnet 1 like this: 192.168.16.7/20, in subnet 2: 192.168.32.7/20, and so on. Note that you need a default router address *in each subnet.*

Marks: One for default router, one for any reasonable strategy.

(c) How will hosts within your subnet above find their IP addresses? [2 marks]

Using DHCP, see Halsall section 6.6.3.
Essentially, we would maintain a database for our hosts. For each we record its MAC (Ethernet) address and its IP address. DHCP looks up a MAC address in the database, and returns its IP address. Marks: One for saying 'DHCP,' another for briefly explaining DHCP.
BOOTP would be another way to do this.

(d) Briefly describe an IPv6 address. Explain which bits are normally used to identify a network and a subnet. [3 marks]

Explained in Halsall section 6.8.2.

- An IPv6 address is much the same as for an IPv4 address, except that it's 128 bits long instead of 32.
- The network number is in the first n bits, n is 32 up to 48. In Halsall figure 6.42 that includes the TLA and NLA fields.
- We use the 'site-level aggregator (SLA)' as our subnet field
- The least-significant 64 bits are our host address.

Marks: one each for network, subnet and host fields (need a little explanation of each).

(e) How are host addresses normally allocated within an IPv6 subnet? What advantage(s) does this offer compared to IPv4? [4 marks]

See Halsall section 6.8.4, Autoconfiguration.

- An IPv6 address for an interface uses the MAC address for that interface. Ethernet MAC addresses have 48 bits, there is a standard method to extend it to 64 bits.
- Every host in a subnet listens for a *Router Advertisement* packet, containing its network and subnet addresses. Routers are usually configured to send RA packets at regular intervals, but a host can broadcast a *Router Solicitation* packet at any time.
- Using IPv6 autoconfiguration provides unique host addresses automatically, thereby simplifying network administration.

Marks: two for explaining IPv6 autoconfiguration (or one for saying *DHCPv6!*), two for any reasonable advantage.