IEEE P802.11
Wireless LANs

Further Simulations Of The Hybrid MAC Protocol

Irvine CA, 9–12th March 1992

Date: 6th March 1992

Source: Mike Smith
Symbionics Ltd
St Johns Innovation Park
Cowley Road
Cambridge
UK
CB4 4WS

Local Reference: W1279/MPS ver 1.0

1. Introduction

This paper describes work carried out to verify the Hybrid MAC Protocol proposed by Ken Biba (Ref 1) and to investigate the performance of the protocol in the presence of channel noise and hidden nodes.

This protocol uses handshaking between two nodes on the network to negotiate for channel bandwidth for the transmission of larger data packets. The protocol has received a lot of interest because of the following reasons:-

- the protocol is inherently simple to understand
- the protocol can be used for synchronous and asynchronous services with almost equal ease.
- the protocol takes into account the problem of hidden nodes.
- the protocol can be said to be tried and tested since it is very similar to the protocol used on Apple networks.

In fact, a version of this protocol is being used for an experimental system (see Ref 2).

2. The Model

The model was constructed using Extend v1.1n graphical modelling package running on Apple Macintosh™ computers. This involved writing custom blocks to perform collision detection, bit-error-rate calculations and to implement the protocol being tested.

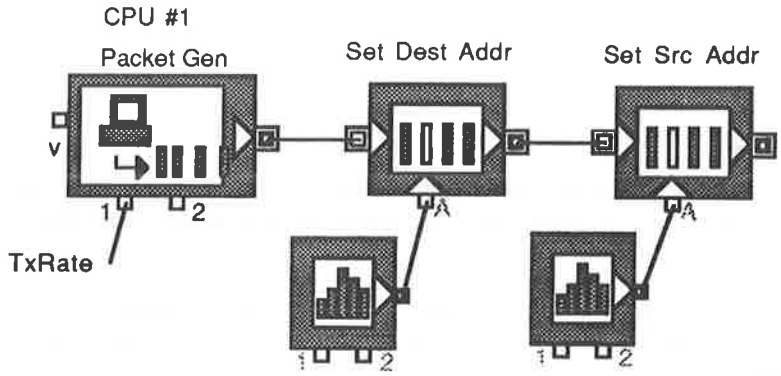
The model is constructed from the following basic functional areas:-

- 10 identical packet generators. Each packet contains a source address, destination address and a length value.
- 10 identical protocol blocks. This block, or set of blocks, simulates the access protocol to the medium.
- blocks to simulate the shared medium. This involves concentrating the packets from all the sources into a single stream.
- blocks to simulate the medium at the receiver. This contains blocks to detect collisions and to apply bit-error-rate calculations.

Each of these areas in the model will be discussed in turn using a diagram of the blocks used taken directly from the model. Any assumptions made will also be described.

2.1 The Packet Sources

The model consists of 10 identical packet generators as shown below.



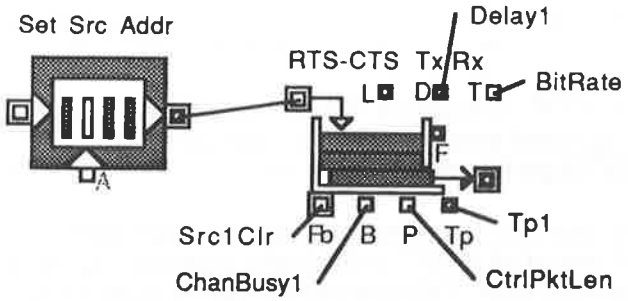
The packet generator generates packets at random time intervals based on a mean interval supplied via the terminal labelled '1'. The distribution used for these intervals is exponential. In the diagram, the mean value is derived from a single point in the model to make it easier to vary the packet rate.

The next pair of blocks give the generated packets a destination address. In this model, the destination address is generated randomly such that each source will randomly select any of the other stations as the destination for the packet.

The final pair of blocks give the packet a length and a source address. The length is generated randomly such that 60% of the packets are 1000 bits long and 40% are 5000 bits long. These values are those used by Ken Biba in his his earlier paper (see Ref 1) and were used to provide some degree of commonality with his results, but other combinations are possible.

2.2 The Protocol Blocks

In the Hybrid MAC protocol, a custom block was written to contain the protocol logic for both transmit and receive. This block is shown below:-



This block handles the transmission of packets received from the packet generator and also listens to all the packets on the network.

The connectors on this block are as follows:-

Fb this input accepts all packets currently on the network.

B the Busy Channel flag. If active, transmission of RTS packets is inhibited.

P an input used to determine the packet length used for RTS, CTS and ACK packets. The model assumes that these packets have the same size.

Tp this output supplies a value which is the achieved bit rate in bits per second. This bit rate is calculated only for DATA packets and gives the network speed perceived by any application interfacing with the MAC and so takes into account the MAC protocol overhead.

T this input connector determines the bit rate used by the system.

D this output supplies the normalised packet delay for DATA packets only. This normalised packet delay is normalised with respect to the packet transmission time so that a delay of 1 means that a packet crossed the network with a delay equal to its transmission time.

L this output supplies the number of packets currently waiting to be transmitted. The internal transmit queue length is limited to 10 packets to prevent memory problems at loads greater than the maximum network load where the transmit queue could be of infinite length.

In implementing the protocol block, the following changes were made to the protocol described in IEEE 802.11/91-92:-

- the CTS, DATA and ACK packets have destination and source addresses. This is needed to resolve ambiguities which can arise in a real network.

  For example, if two stations transmit an RTS packet simultaneously to a third station, if the receiving station only receives one of them, it will transmit one CTS packet. Since both senders will see this packet in the protocol as described in IEEE 802.11/91-92, both senders would believe that they were being requested to send a DATA packet, so causing an immediate collision. By using a destination address both stations can determine who is being permitted to transmit. This change means that the RTS, CTS and ACK packets will be the same size.

- when an RTS packet is sent by a station, it waits until a CTS frame is received, an RTS from another station is heard or the retransmission timer times out.

  When another RTS is received, the transmitter desists from transmitting until the end of the requested time slot. Then the station will retransmit after a random delay of up to M RTS packet transmission periods.

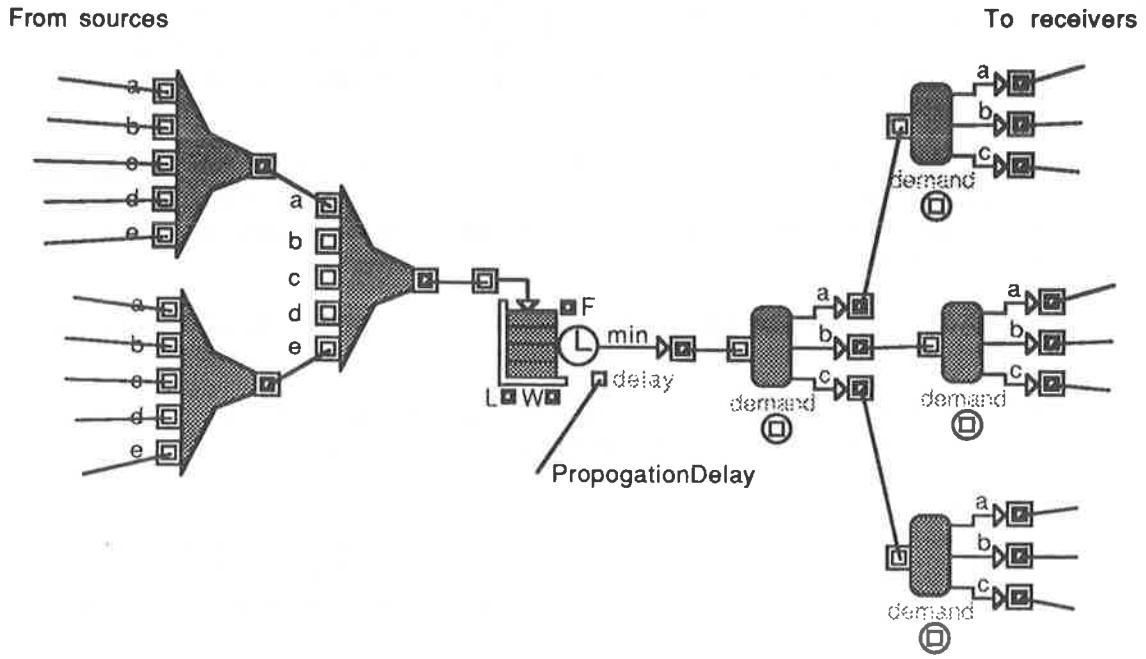
  If no CTS is received and there is no other channel activity, the RTS is transmitted after a fixed period of N RTS periods plus a random delay of M RTS packet periods. The fixed part of the retransmission period is allow time for the receiving station to reply. The variable part is needed to prevent two stations, attempting to send RTS packets simultaneously, from colliding again.

- when a DATA packet is sent by a station, it waits for a period of time equal to the transmission time of the DATA packet plus N times the transmission time of the ACK period. If no ACK is received after this time, the station will attempt to retransmit the DATA packet starting with the RTS packet. Because the transmitter will attempt transmission soon after the end of its time slot, the station is effectively given priority over other stations.

In a real system, the number of retransmission attempts for a given DATA packet would be limited. In the current model, M is set to 10 and N is set to 2 and retransmissions are carried out until the DATA packet has been delivered.

2.3 The Shared Medium

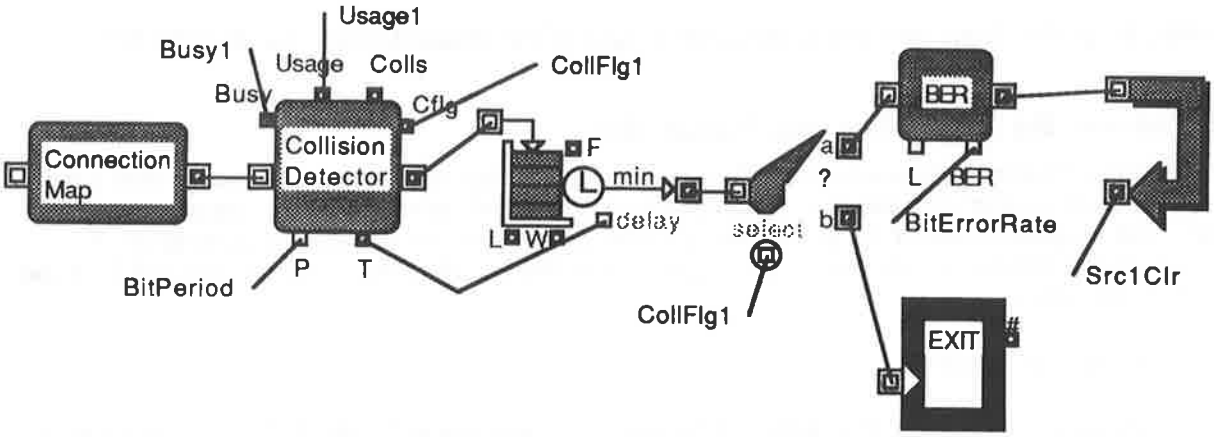
In the model, the medium is represented by block used to concentrate the packets into a single stream and then to generate copies of each packet for the receiver section of each station.



The first three blocks combine the output from the packet generators into a single stream. The packets are then delayed for a short time to simulate the network propagation delay. Finally, each packet is duplicated and a single copy is sent to each receiver (NOTE: to fit the diagram onto the page, one of the outputs from this block has been omitted).

2.4 The Receivers

The following set of blocks attempts to simulate the receiver hardware in a station and also takes care of determining which stations are in range.



The Connection Map block contains a n by n matrix which determines the connectivity between nodes. In the current model, with 10 nodes this map contains 10 rows and 10 columns where rows are used for destination nodes and the columns for source nodes. Each entry in the map can take a probability value between 0 (for stations which cannot be heard by this receiver) to 1 (all packets from this station can be received). Since there are two entries in the table for all possible pairs of nodes, the probability that a packet can be heard over the paths A->B and B->A can be different. On receiving a packet, the Connection Map block extracts the source address and calculates whether the packet is to be passed on. If not, the packet is discarded. Each receiver has a copy of the same connection matrix.

The Collision Detector block looks at the arrival time and packet length to determine collisions on the network. The block also provides outputs which an be used to control other blocks in the simulation. This block passes packets on without delay. This block has the following inputs and outputs:-

Busy this output is set to 1 when there is a packet being transmitted over the network and 0 when the channel is quiet.

Usage this output provides a value equal to the channel utilisation as a value between 0 and 1 where 1 represents 100% utilisation.

Colls this output supplies a value equal to the number of collisions detected.

Cflg this output is set to 1 when there is a collision on the channel and 0 otherwise.

T this output provides the transmission time for the packet passing through the block.

P this input supplies the block with the current bit period (1/bit-rate).

The next block delays the packet by a time equal to its transmission time. The delay is derived from the T output on the Collision Detector block.

The next block is a switch used to remove collided packets from the system. The switch is controlled by the Cflg output from the Collision Detector.

Packets which have been not collided are passed to the Bit Error Rate block. This block uses the current bit-error-rate to determine the probability the one or more bits in the packet has been lost. If the packet contains a bit error, this block discards the packet such that it is not received by the protocol block.

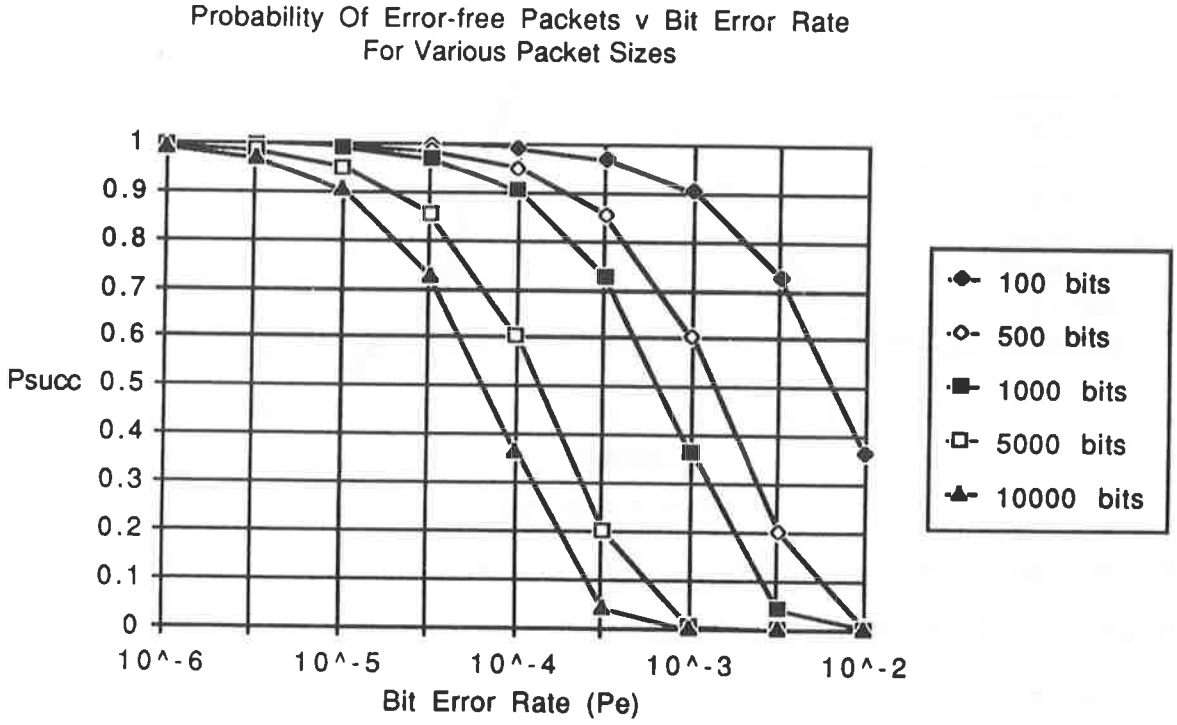
Finally, the packet is passed back to the receive input of the protocol block discussed earlier.

2.5 Physical Bit Error Rate and Packet Size

The received bit-error-rate places limitations on the packet sizes that can be transmitted over a radio LAN. If no forward error correction is used, there is a trade off between the maximum packet length and the probability of the packet being propagated across the network without error. If Psucc is the probability of successful propagation, L is the length of the packet in bits and Pe is the bit-error-rate, then

$$Psucc = (1 - Pe)^L$$

This equation assumes that the bit errors occur randomly and not in bursts. If Pe and L are varied, the following set of curves is obtained:-



This chart shows that , with no error correction, the system needs to have a bit-error-rate of better than 10^{-4} if Ethernet-size packets are to be used (up to 1500 bytes or 12000 bits).

In the simulations we have carried out, data packets of 1000 and 5000 bits are used. The chart above indicates that with these packet sizes, the simulation will work with bit-error-rates better than 10^{-4}. With bit-error-rates worse than 10^{-3}, the protocol fails to operate because the large data packets have a very low probability of being successfully transferred.

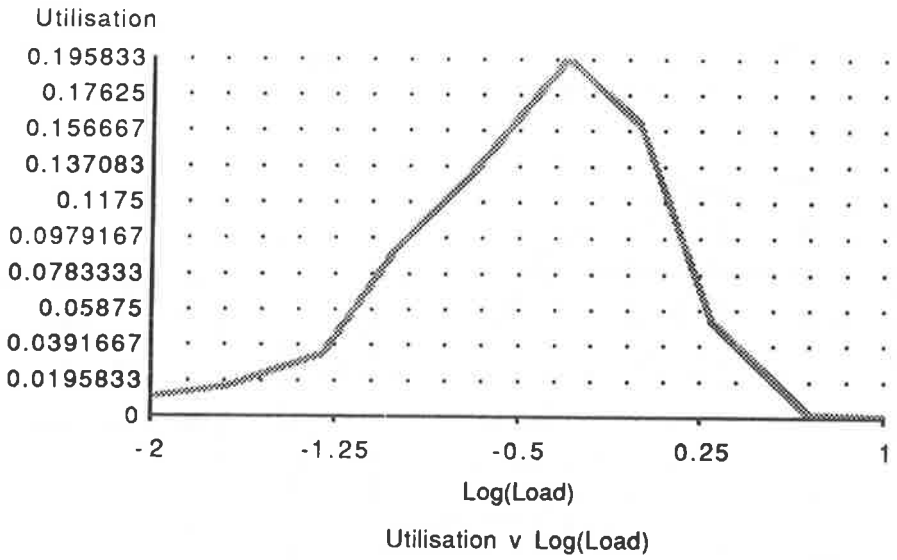
2.6 Hidden Nodes

In the models used in this paper, hidden nodes are simulated by setting up the Connection Map blocks such that the path between nodes has a limited reliability. If a node only has a 95% chance of hearing packets from any other node, this implies that 5% of the nodes on the network are hidden from one or more of the other nodes. Because the connection map contains probabilities, the connectivity of the network is constantly varying as is likely to be the case in a real network.

3. Reference Protocols

To check the validity of the simulation system, models were constructed for some textbook protocols to check that the results obtained are correct. These protocols were the Aloha and the Slotted-Aloha protocols. The simulation results were taken directly from Extend.

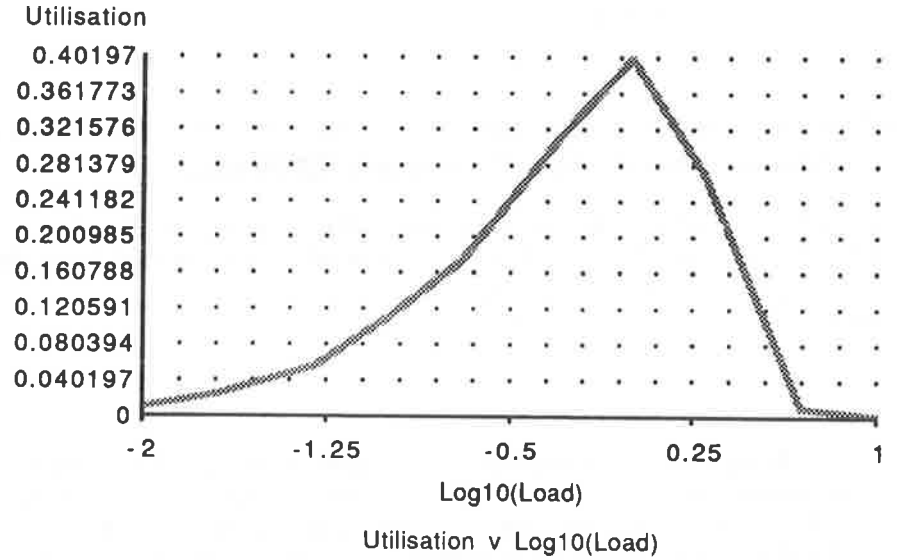
The channel utilisation versus channel load curve for the Aloha protocol obtained is as follows:-



Utilisation v Log(Load)

This curve agrees with the theoretical curve with reasonable accuracy.

The same curve for the Slotted-Aloha protocol is:-



Utilisation v Log10(Load)

This shows the expected doubling in utilisation with respect to the Aloha protocol as predicted in theory.

Both of these simulation were carried out with the following parameters:-

Transmitting stations:	10
Bit-error-rate:	10^{-6}
Propagation delay:	10 μs
Packet length:	1000 bits

4. The Hybrid Asynchronous MAC Protocol

The following results are for the Hybrid MAC when operating asynchronously. The results confirm those obtained by Ken Biba and also include simulations carried out to look at the protocol performance under conditions likely to be found in actual networks.

In each case, the utilisation and delay curves are given but in addition, a graph of achieved network throughput as perceived by a user is also given. The latter curve is more important to someone considering the purchase of a wireless LAN and it also shows the impact made by MAC protocol overheads on the overall achievable network throughput.

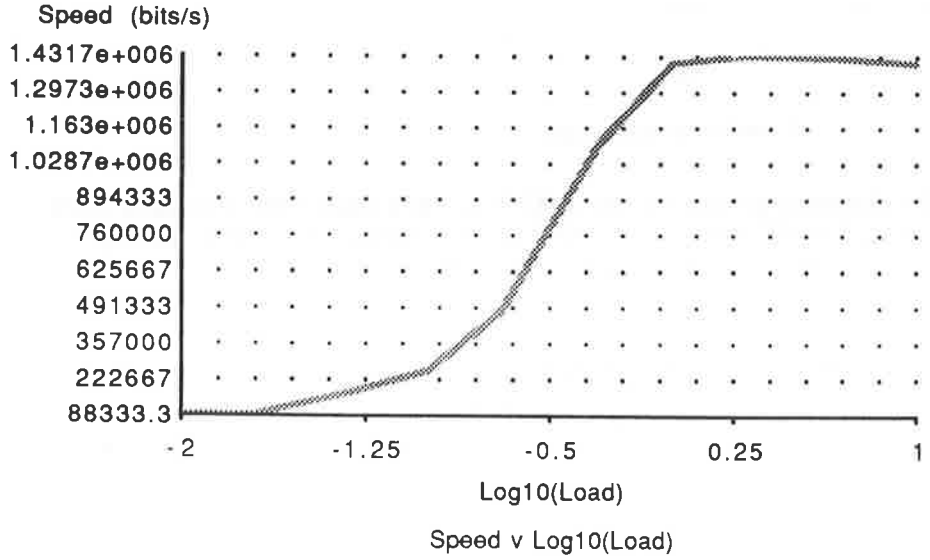
It should also be noted that in a wireless LAN, each receiver will perceive different network utilisations. For this reason, it is possible for the overall network utilisation to exceed 100%, particularly when hidden nodes exist.

The delay curves given show a maximum delay of 50 packet periods. This is an artificial limit imposed to ensure that the curve can be plotted. A delay of 50 packet periods is actually an infinite delay.

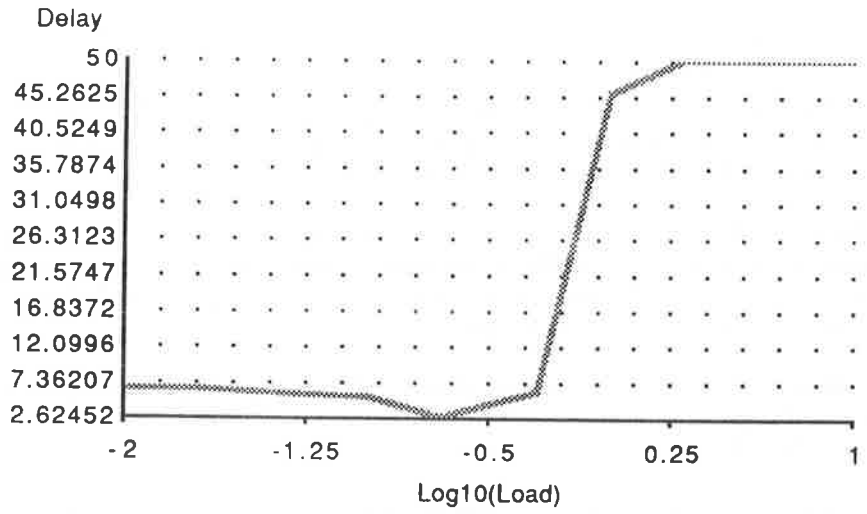
4.1 MAC Performance With Error-free Channels

This simulation was carried out with the following parameters:-

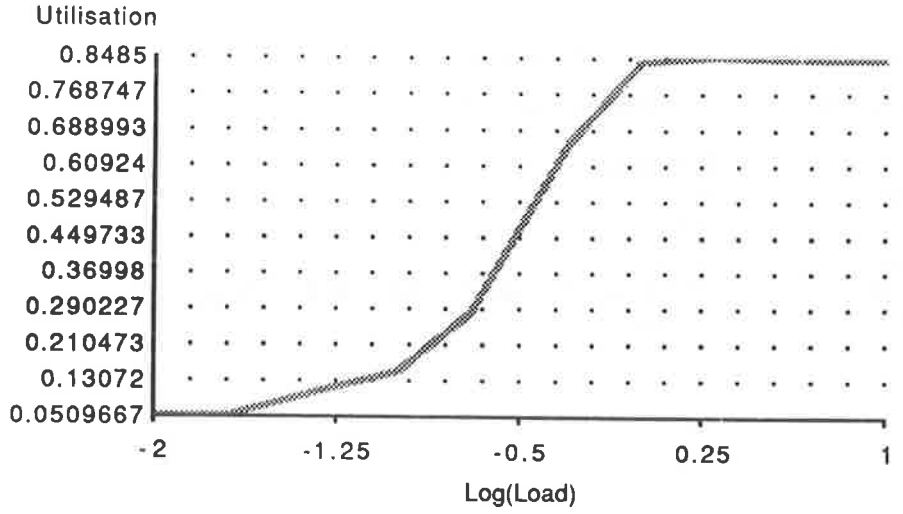
Bit rate:	2 Mbits/s
Transmitting stations:	10
Bit-error-rate:	10^{-6}
Propagation delay:	10 µs
Mean Packet length:	2600 bits (60% @ 1000 bits, 40% @ 5000 bits)
Control packet length:	160 bits



Speed v Log10(Load)



Delay v Log10(Load)



Utilisation v Log(Load)

These results show that the channel utilisation peaks at 85% at 100% load. This utilisation level agrees with the result presented in IEEE 802.11/91-92. The throughput curve shows that the achievable network bit-rate as seen by the user is approximately 1.4 Mbits/s.

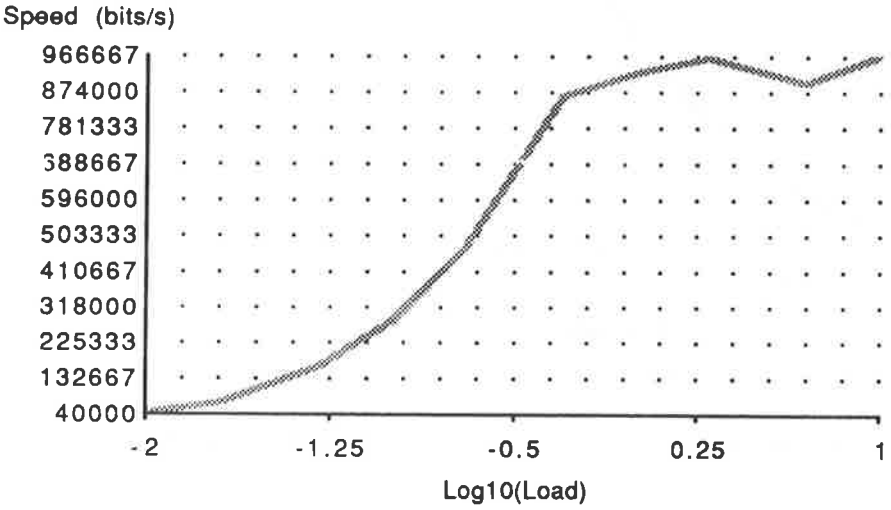
4.2 MAC Performance With Error-prone Channels

This simulation was carried out with the following parameters:-

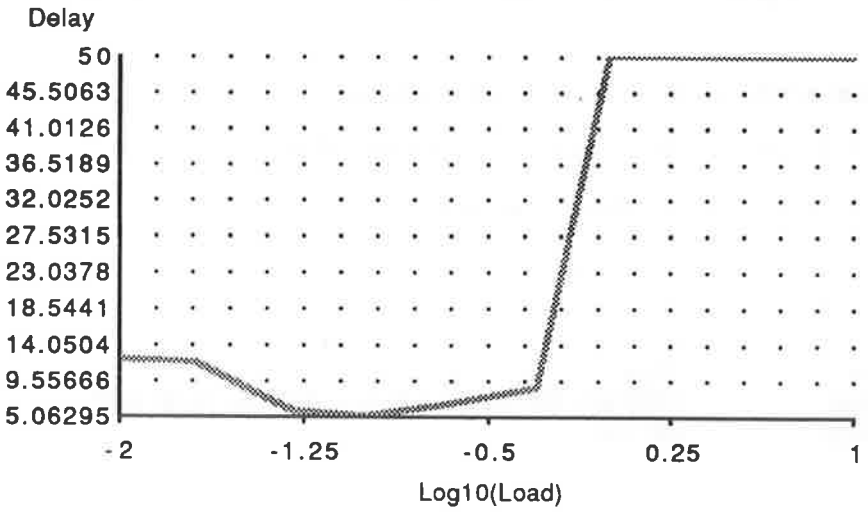
Bit rate:	2 Mbits/s
Transmitting stations:	10
Bit-error-rate:	10^{-4}
Propagation delay:	10 μs
Mean Packet length:	2600 bits (60% @ 1000 bits, 40% @ 5000 bits)

Control packet length: 160 bits

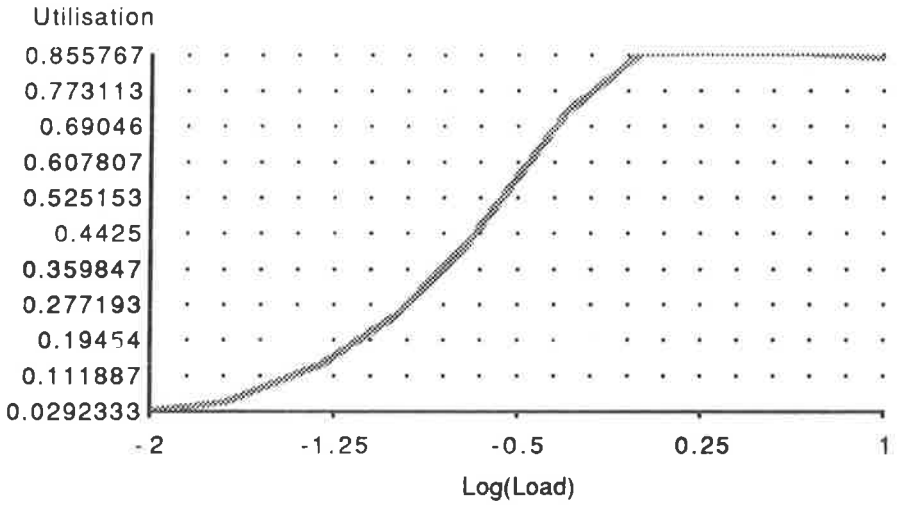
The bit-error-rate for this simulation represents the limit for the protocol as it stands at the moment since it allows the 5000-bit packets to pass through the network with a 20% probability of success.



Speed v Log10(Load)



Delay v Log10(Load)



Utilisation v Log(Load)

Because of the way the model measures utilisation, these results show the same channel utilisation of 85% at 100% load.

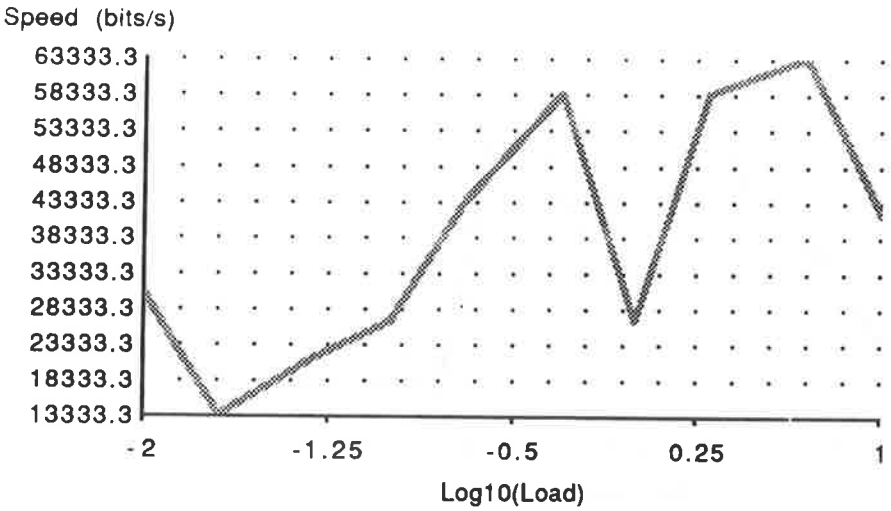
However, the throughput curve shows that the achievable network throughput has degraded to approximately 0.97 Mbits/s. This is due to the retransmission of data packets required because of packet corruption. The control packets suffer less from errors because of their smaller size.

4.3 MAC Performance With High Error Rate Channels

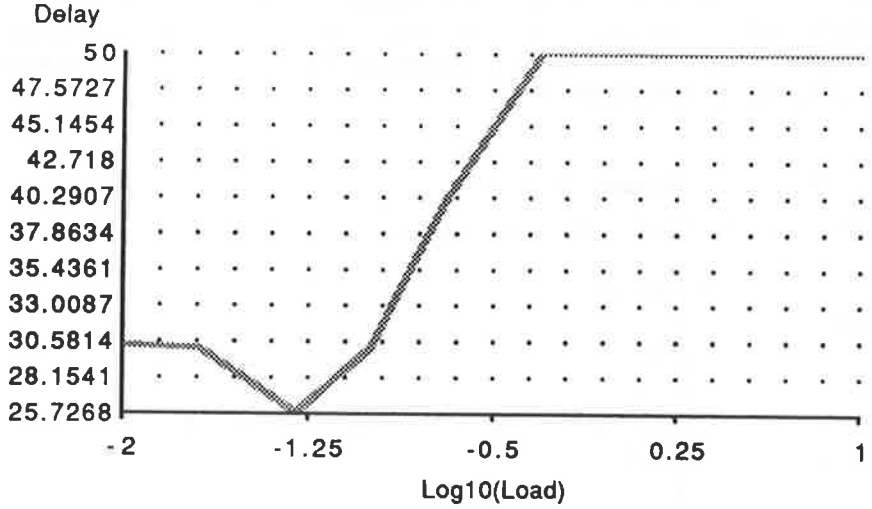
With the current protocol, a high error rate is any error rate worse than 10^{-4}.

This simulation was carried out with the following parameters:-

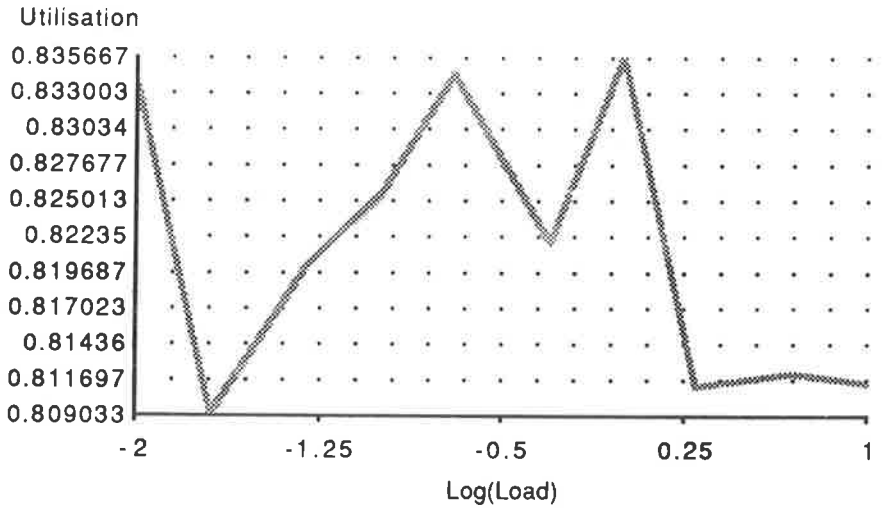
Bit rate:	2 Mbits/s
Transmitting stations:	10
Bit-error-rate:	10^{-3}
Propagation delay:	10 µs
Mean Packet length:	2600 bits (60% @ 1000 bits, 40% @ 5000 bits)
Control packet length:	160 bits



Speed v Log10(Load)



Delay v Log10(Load)



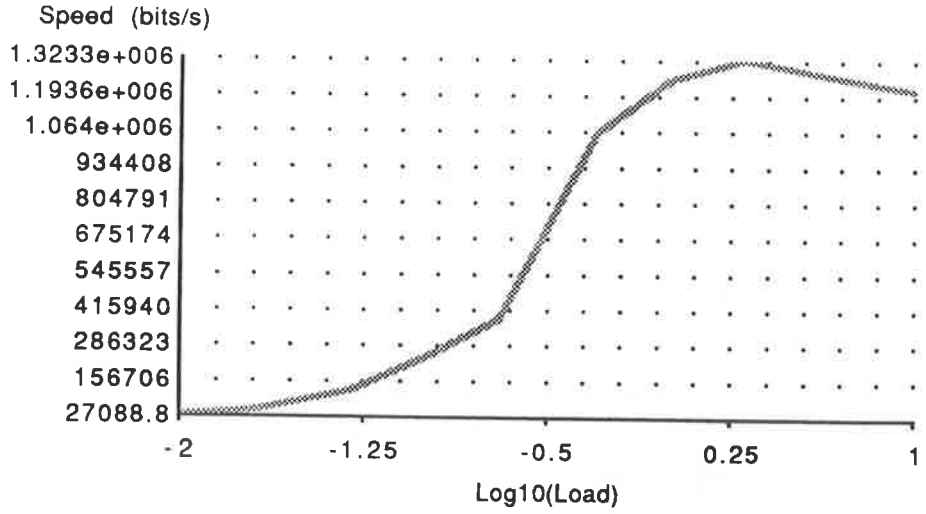
Utilisation v Log(Load)

These curves show that the achieved network bit rate has dropped to a very low level. This is due to the fact that very few 5000-bit data packets are being transferred across the network without error, although the 1000-bit packets have a 35% chance of success. This means that the protocol is spending most of its time carrying out retransmissions.

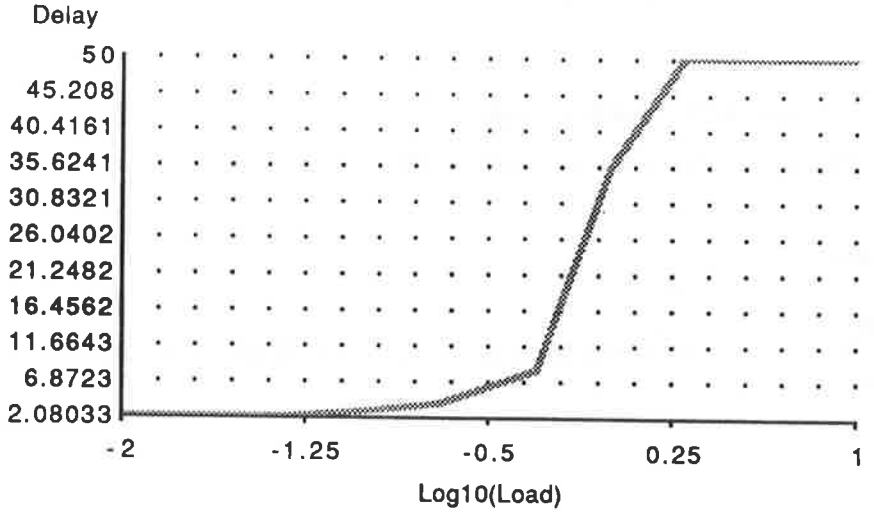
4.4 MAC Performance With 5% Hidden Stations

This simulation was carried out with the following parameters:-

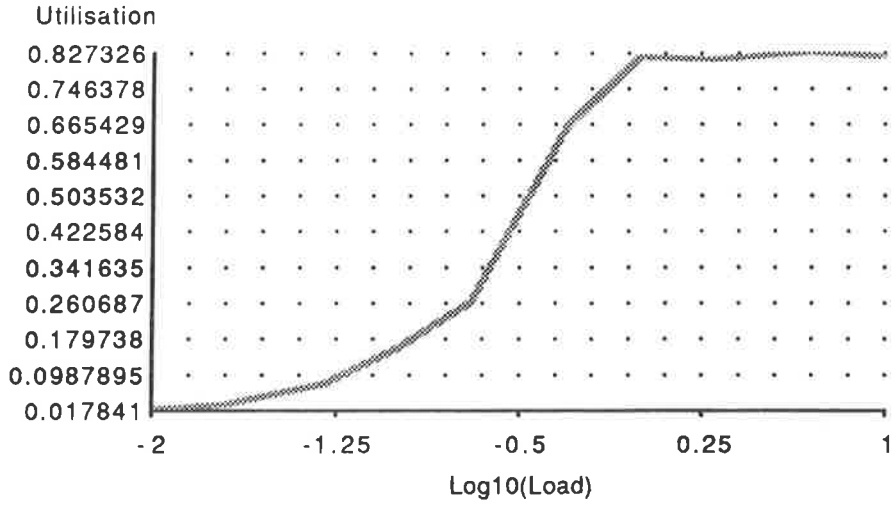
Bit rate:	2 Mbits/s
Transmitting stations:	10
Bit-error-rate:	10^{-6}
Propagation delay:	10 µs
Mean Packet length:	2600 bits (60% @ 1000 bits, 40% @ 5000 bits)
Control packet length:	160 bits
Hidden stations:	5%



Speed v Log10(Load)



Delay v Log10(Load)



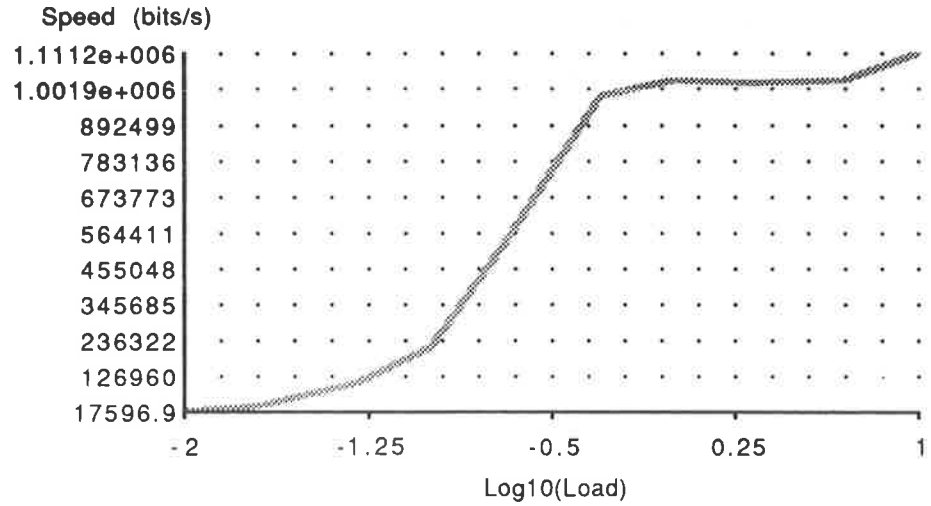
Utilisation v Log10(Load)

These results show that there is only a slight degradation with 5% hidden stations.

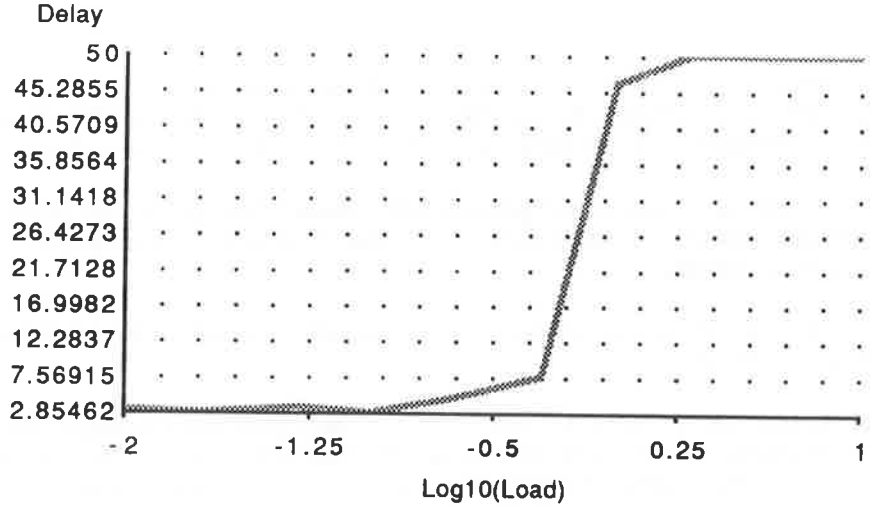
4.5 MAC Performance With 10% Hidden Stations

This simulation was carried out with the following parameters:-

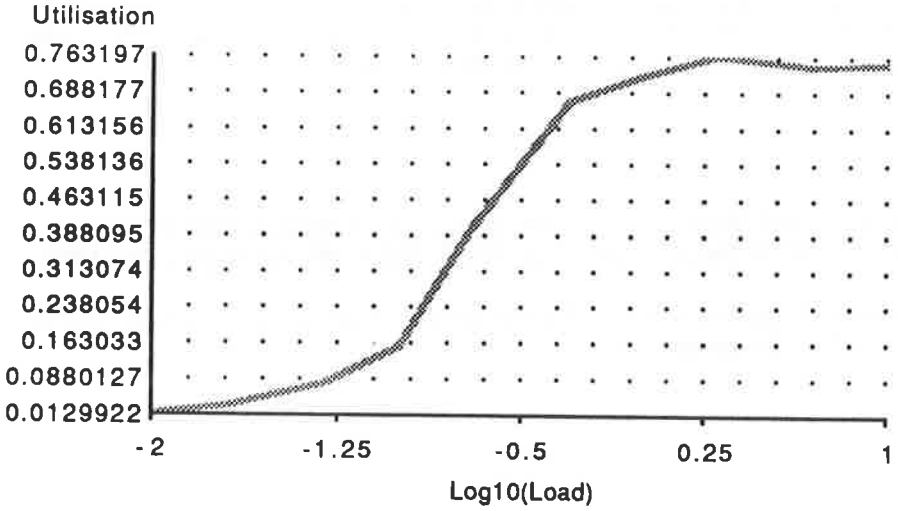
Bit rate:	2 Mbits/s
Transmitting stations:	10
Bit-error-rate:	10^{-6}
Propagation delay:	10 µs
Mean Packet length:	2600 bits (60% @ 1000 bits, 40% @ 5000 bits)
Control packet length:	160 bits
Hidden stations:	10%



Speed v Log10(Load)



Delay v Log10(Load)



Utilisation v Log10(Load)

With this level of hidden stations, there is about 30% reduction in the maximum network throughput caused by the increased collision rate.

5. Conclusions

From the results above, it can be seen that the protocol simulated is quite robust and copes quite well with bit-error-rates of approximately 10^{-4} and better and with up to 10% hidden nodes. At the time of writing, the full results were not available for 25% hidden nodes but preliminary figures indicate that utilisation falls to 54% and the maximum throughput falls to 600 kbits/s.

The major problem with the protocol at the moment is its poor performance when transmitting large, Ethernet-sized packets over noisy channels with physical bit-error-rates worse than 10-4.

Since bit-error-rate depends upon receiver performance and the modulation techniques used, it is difficult to say what physical bit-error-rates should be expected. For example, infra-red systems are likely to offer a lower physical bit-error-rate than radio systems.

For the sake of flexibility, we should assume the worst case (i.e. radio) and design the MAC protocol to cope with all expected conditions. To this end, the following possible modifications to the protocol would allow its performance to degrade more gracefully over error-prone channels:-

- use forward error correction on the packets transmitted. This adds overhead to the packets and there may be an upper limit on the size of packet to which error correction can be applied.

- gradually reduce the MAC data packet size as channel noise increases to increase the chance of error-free transmission.

If the MAC protocol could determine the current bit-error-rate, it could dynamically apply one or both of these techniques dynamically under noisy conditions. This would allow the protocol to work efficiently under near-error-free conditions, but also to degrade more gracefully in the presence of noise.

The problem with hidden nodes cannot be readily improved. However, it is worth pointing out that in a real network, stations will eventually give up attempting to transmit to a station if it is hidden from radio contact for a finite period of time. This means that in reality, hidden nodes will probably cause less affect to network performance than the model above implies.

6. References

1. Ken Biba, A Hybrid Wireless MAC Protocol Supporting Asynchronous and Synchronous MSDU Delivery Services, IEEE 802.11/91-92, September 1991

2. Peter Cripps, Engineering Choices for Portable Wireless LAN Adapters, IEEE P802.11/91-122, November 1991