Reply to Examiner No. 2

From Candidate:	**Erwin Anggadjaja**
Degree:	**Doctor of Philosophy**
Thesis Title:	**Cross -layer MIMO-link Exploiting Packet Switching and Adaptive Modulation for TCP/IP Enabled Volcano Monitoring Networks**

Chapter 1

Discussions as expected.

Chapter 2

Discussions as expected.

Chapter 3

1. The text surrounding Fig. 3.2. does not explain how the receiver determines that the previous ACK has been lost – this omission should be remedied.

   Fig. 3.2 in the thesis illustrates the basic concept showing how the re-ACK works in the model. In our model, the ACK process is run as follow. Once the packet is received correctly, the receiver will send an ACK back to the sender and the same time it activates the ACK-timer. Once this value expires – please refer to the Fig. 1 below – (indicated when *timeoutEvent_ACKRetry_Y11* is up), the receiver will assume that the (previous) ACK has been lost. So it will re-send the same ACK number packet (*DupACKPOP2*) with newly ACK-timer value (*timeout_ACKRetry_Y11*). Note that in the model, the ACK-timer value is adjusted so it will not trigger a resending packet routine from the transmitter.

   The text in the thesis has now been improved to make this point clearer.

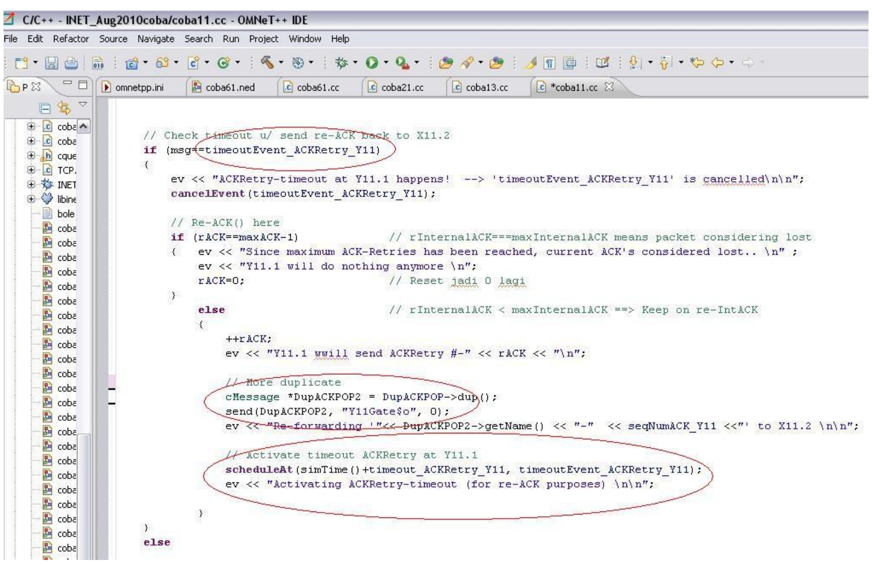

Fig. 1. Snapshot of the code showing how re-ACK is coded.

2. Segment length is called 'window size' (w) instead of the usual term 'block size'. Window size is normally used for the number of blocks or packets comprising a "window" in a data link or TCP connection

   Thank you for pointing this out. At the beginning, the term 'window size' is used as a parameter for coding purposes. It is true that this might be confused with the 'window size' term of TCP. To reduce the confusion, the use of this term has been changed in the revised thesis.

3. The size of ACKs is not mentioned

   The size of ACK used in all simulations is set to 32 bytes. However, for easier implementation, the model assumes that all ACKs experience no bit error. In practical implementation, the ACKs can be resent several times, and it has been shown in the paper by Prakash and McLoughlin [1] that for the BER range explored in this thesis, this method of automatic multiple signal packet resends results in a virtually zero error rate (in excess of 10^{-12}) with negligible overhead.

   [1] Prakash S., McLoughlin I. (2011). Effects of channel prediction for transmit antenna selection with maximal-ratio combining in Rayleigh fading. *IEEE Transactions on Vehicular Technology*, 60(6), 2555{2568.

4. Another important metric, latency, is not considered – the effect of significant wireless-link time-delay should be discussed.

   Thank you for pointing this out. This issue is addressed jointly with the comments for Chapter 4 question no. 3, below.

5. A shortcoming is that NAKs (negative ACKs) are not employed, as in a practical Radio Link Protocols (RLP), which would avoid the time delay and bandwidth overhead of an ACK-based protocol. A discussion on what benefits the use of NAKs could bring should be added.

   Actually, the Radio Link Protocol (RLP) can make use of either ACK- or NAK-based error control mechanisms. In a NAK-based implementation, the sender will assume that all the frames are sent successfully as the receiver merely replies with the un-successful / out-of-order received frames as a NAK. Should this mechanism is implemented in the model described in this thesis, the benefit that could be achieved are:

   1. Improvement in channel's efficiency, as fewer frames will be sent over the air.
   2. Minimizing the transmission latency (thus improving effective throughput) since the system does not need to wait for 'unnecessary' acknowledgement-frames which ACK-based protocol does. Although all received and reconstructed frames would need to wait for the maximum NAK period before being output.

   As in this thesis, the work does not use the NAK protocol for two main reasons. First is to minimize the computation complexity in the receiver. All the frames successfully received are guaranteed correct in-order to ease the progress of reassembling. Second, the NAK-based protocol implicitly indicates the need of a significant output buffer to facilitate the correct reassembly and assumed-acknowledged out-of-order frames. By directly implementing the reassembly process as this thesis does, and then immediately outputting known good packets, this issue can be disregard.

To put this another way, yes, the candidate is aware that NAK has certain advantages, and it is probably one implementation choice that might be made in practice. However the focus of this thesis is on the effect of *other* methods of improving goodput based on packet handling. Since this work concentrated on these, the candidate implemented the simplest possible working baseline system for testing. In fact, the improvements (or degradation) due to different methods of packet handling are independent of NAK/ACK method. The basic results hold for both classes of system. The revised thesis now notes this fact.

<u>Chapter 4</u>

1. This chapter presents TCP and UDP, and the drawback over wireless links, and some survey of previous TCP enhancement. But none of the methods are employed in the remainder of the thesis. The candidate was trying to ensure that suffice explanations about basic of transport protocol and surveys on the recent TCP improvement is covered. If this chapter was missing, definitely one examiner would ask, “Why didn't the candidate consider the various other methods of enhancing TCP for a wireless link?”

   We also needed to establish that none of these methods works really well or is a viable one-size-fits all solution. The candidate then considered re-visiting and adopting the basic TCP mechanism, but enhancing its handing over wireless with novel improvements such as packet switching mechanisms, re-routing methods, and lastly adaptive modulation (without altering the standard TCP compatibility of the system).

2. The Error Control mechanism dispenses with re-ACKs (unlike in Chapter 3) and only employs re-tries after timeouts, but NAKs after an out-of-sequence arrival may have been a better solution. As discussed earlier, it is true that using the NAK protocol could benefit the system. However please bear in mind that with multiple sub-channels being employed each with different BERs (and later, each with different bit rates), the number of out-of-sequence arrivals will definitely increase. Much better to buffer out-of-order packets in the reconstruction buffer with a positive ACK confirmation before emptying the reconstruction buffer and outputting the received data.

3. Again, delay metric is not considered.
   Latency describes a time-delay experienced in a system measured as either a one-way or round-trip measurement. The main source of latency here is propagation delay, which describes how long actually the time needed to transfer packet between sender and receiver given the erroneous receipt of data. In addition to that, latency is closely related to several causes. Some of them are rate of transmission, packet size, and most likely the protocol used in the communication; also possible to include a routing mechanism, nodes hop-length, as well as queuing/buffer.

   In the thesis, obviously, the latency mainly is affected by the protocols employed. The first system employed the SAR-ARQ protocol (retransmission scheme), followed by the TCP mechanism (mainly congestion window and fast retransmit and recovery), and lastly applying an AM adjustment. Thus each of these implementations will potentially add to the system’s latency.

The effect of this latency, noticeably, will be in the performance of the system, i.e. in the form of throughput/goodput – which is the evaluation metric used in the thesis. Subsequently, though this thesis did not observe latency itself as a main (or discrete) metric, its effect can be observed by the goodput performance of the system.

Related to this issue, the candidate also found two interesting works:

- The use of hybrid FEC and ARQ in TCP [2].
  It is known that the TCP protocol encounters latency issues as (a) its throughput is inversely proportional to the average round-trip time [1] and (b) the possibility that the retransmission time-out timer may expire when the error control (congestion window) mechanism is still in progress. A preliminary work by [3] and [4], proposed a mitigation scenario by using a combination of FEC and ARQ. The motive was simple: FEC can resolve those two problems mentioned above as it does not cause an increase in RTT and time-out timer. The hybrid FEC-ARQ scenario was implemented, and as the result it showed that the scheme is able to carry more traffic during the minimal latency - thus increasing the effective throughput.

- The use of an 'upgraded' transport protocol, such as Multipurpose Transport Protocol (MTP), which is developed by Data Expedition, Inc (DEI).
  A combination of TCP and UDP is adopted, with the former is used to guarantee the integrity of data while UDP is chosen for speed and utilization; MTP is designed so that it runs on top of UDP as a proprietary substitute to TCP. It uses a more efficient flow-control algorithm. As for the results, it is claimed that, after installing the protocol (software) development, the throughput for mobile devices increases significantly, i.e. achieving 42-45 Mbps , from 6-7 Mbps [5]. Though the idea is thought-provoking, the candidate did not find any subsequent literature and work on this protocol. The latest was found in 2011. If low latency is a major requirement in a particular deployment scenario, this may be a promising way to achieve it.

[1] C. Barakat, E. Altman, and W. Dabbous, "On TCP performance in a heterogeneous network: a survey", IEEE Communications Magazine, vol. 38, no. 1, pp. 40 – 46, 2000.
[2] R. Abdelmoumen, M. Malli, and C. Barakat, "Analysis of TCP latency over wireless links supporting FEC/ARQ-SR for error recovery", IEEE Int. Conf. Communications, vol. 7, pp. 3994 – 3998, 2004.
[3] M. Allman, D. Glover, and L. Sanchez, "Enhancing TCP over satellite channels using standard mechanisms", RFC 2488, Jan. 1999.
[4] H. Balakrishnan, V. N. Padmanabhan, S. Seshan, and R. Katz, "A comparison of Mechanisms for Improving TCP Performance over Wireless Links", ACM SIGCOMM, Aug. 1996.
[5] http://www.cablefax.com/tech/sections/features/Solving-The-Slowdown-Ways-To-Reduce-Wireless-Network-Latency_45347.html

4. When deciding maximum number of retries (r_{MAX}) = 4 for the range of segment size (SS) and BER, no insight is provided as how widely applicable is the result.
   A justification was given in the thesis in terms of deciding the best r_{MAX} value. Having r_{MAX} value less than 4, means that a reduced goodput is noted under the experimental conditions. On the other hand, for a value bigger than 4, though higher goodput is achieved, it will increase the possibility of having longer transmission time as it allows more packets to be retransmitted over the air. Clearly there is a trade off here, but for the purpose of testing, the candidate decided that r_{MAX} = 4 is the best in the given scenario (and this is both discussed and concluded inside the thesis).

When considering the wider system applicability, in this case, allowing extra packet transmissions means tolerating the resulting extra power consumed by the system, as well as potentially increased latency. Under a requirement of reducing power consumption, it would not be good to extend r_{MAX} too far. For others wishing to implement this system, testing the effect of changing r_{MAX} is relatively easy. The effect his highly application dependent, so the thesis only contains a general description of these effects. However the discussion has been clarified and improved in the revised thesis.

5. In Fig. 4.12, the goodput seems to increase with SS at lower BERs, but decrease with SS at highest BER. Discussion should be added to address this.
   The graph in Fig. 4.12 (now Fig. 4.11) shows a comparison of average goodput of selected channel conditions for all segment size (*SS*) values. Referring to Equation (4.2), the goodput is linear with the number of correct transmissions. For channels with high error rate, the number of received packets will be low due to the discarded erroneous-packets. As this value is low, a lower goodput value is yielded, compared to the goodput in the channel with a low error rate. This explanation has been added to the revised thesis.

<u>Chapter 5</u>

1. Details of the SWS and PLD scheme, such as how TCP packets are interleaved to avoid idle slots, are omitted – material should be added to address this omission.
   Referring to Fig. 4.7, the data is delivered through smaller packets named 'TCP payload'. In the OMNet++ model of the system (Fig. 4.4), these 'TCP-payloads' will be sent to Node X, to be processed with a segmentation mechanism and then go through the switching schemes. During the scheme, the next-'TCP payload' packet will be stored in the sender buffer of Node A (Note that the system utilized the full TCP mechanism, including the implementation of sending window/buffer). Once the previous packet is safely delivered, then this subsequent packet will be processed right away. The free transmission slots are filled on a first-come first-served basis, following the payload routing/switching rules. The assumption is made that sufficient TCP packets are being sent so that the transmission buffer does not run empty in practice. Also that reconstructed packets can be output instantaneously without waiting for output buffering from the output of the receive node. Payload switching generally only comes into play during resends, in which case there will be an inevitable delay in the resent payloads with respect to correctly transmitted ones.

2. The model for performance analysis assumes that the two links have the same long-run average BER, but a perhaps unrealistic assumption is made that the BERs are negatively correlated rather than being random – a justification for this based on practical considerations should be added.
   It's a good point! Some studies in multi-antenna system showed that channel characteristics are often unequal [1], for example in a two-channel system, one typically has higher BER than the other. In such systems, different levels of BER can be observed in each of the channels, i.e. one can simply categorize the channel (good or bad) - and these values are also changing from time to time.

It does not mean that one channel automatically improves when the other degrades, but simply that at any one time instant, one channel would naturally be better than the other – and this is the case in practice. There is no correlation here between bit errors in independent channels. Everything is calculated on a frame-by-frame basis, and over a single TCP packet transmission time and above, there is no correlation made between channels: they have entirely independent errors. The only constraint is that the long-term error rate is fixed – and again this is what would happen in practice. In practice we would not know the precise long-term BER value until AFTER a transmission run. But once the run had finished, we would characterise its performance by its average BER.

More on this, a practical test by [2] (cited in the thesis as reference [13]) showed clearly that when exploring 2-channel statistic of a TR-STBC (time-reversal space-time block coding MISO) system, it is found that:

1. The distribution of BER of each channel is variably spread around the mean value.
2. It can be easily spotted that one channel is usually good while the other is usually bad in a real system. It could be because of antenna placement, the impedance of RF feed wires, inefficiencies, or drift in the RF circuits – at both transmit and receive end. This is all discussed in the thesis – the explanation for this in the revised thesis should now be clearer.
3. The BERs value in the channels show certain differences, but their averaged performance is identical.

   This means that in a real deployed system, it would experience time-varying instantaneous BERs according to the statistical properties of the transmission over the particular wireless channel.

The above findings are the fundamental motivation in this thesis to use the BER statistic in modelling the multi-channel system. It is very normal! In fact, the authors of that paper believe that it is far more 'normal' for the channels to be slightly different than it is for them to always be exactly the same. And yet the vast majority of simulations tend to assume a fixed BER equally distributed between channels.

Again, as mentioned, the revised thesis now explains this point more clearly. Since the examiner brought up the topic, the opportunity has been taken to now discuss it more extensively in the thesis.

[1] I. McLoughlin & K. Mehrotra, "Time reversal space time block coding with channel estimation errors", *Int. Conf. on Integrated Circuits and Systems*, Singapore, Dec. 2003.
[2] I. McLoughlin and H. Sirisena, "TCP/IP link layer error mitigation for MIMO wireless links", *Telecommunication Systems*, Volume 50, Issue 3, pp 137-148, 2012.

3. An improvement is next proposed where re-tries are sent on a different link, either the alternate one or one chosen randomly, to the one that experienced the error. The details of the interaction with segments of other TCP packets on the same links are omitted – this should be elaborated upon.
   The interactions between segments on the same link are:

a. Independent

• Each segment is transmitted independently. For example whether segment (*a1*) is successfully received or not in particular link, it does not affect the subsequent segments – except when it has reached the maximum retries value for that packet.

• While a segment is in the transmission process, other segments are waiting in the available buffer (at Node X). Note that, the system does not overwhelm the memory buffer since the buffer is limited based on the number of the block sizes created.

b. set apart from the sequence number (*SN*).

Node X would be able determine -for each block size- which link to transmit based on its sequence number. It uses a simple computation coding to make sure each block size has a fixed link-number - whether it would be transmitted into other or random link.

The thesis has been amended to make this clearer to the reader.

4. The test results lead to the conclusion that the combination of PLD with deterministic link selection, called P-RDL, is the best approach under the test conditions. But whether or not this remains true under different conditions remains an open problem. Discussion of the sensitivity of the findings to the assumed operating conditions should be added.

   Although we cannot test every scenario in practice, this thesis has tried to investigate the behaviour in a few common operating modes. The conclusion holds true for the BER range and operating scenario discussed.

   In terms of BER sensitivity, as this improves, and eventually becomes very good, there would not be any need to implement P-RDL: it would give no (or at best very marginal) improvement. Conversely, at the extreme end of 'bad', performance halts irrespective of which method we choose. However we know (from the experimental results already reported) that P-RDL performs best in the intermediate region.

   To assess another possibility, the candidate has added another scenario (added into Section 5.3.6) to the investigations. This time, Link A is 'forced' to consistently get worse (instead of getting better as used in all previous simulations), while the remaining links improve. The results are shown in Fig. 2 and Fig. 3 below.

   In fact, this result shows that the new scenario yields S-RRL as the best approach, slightly outperforming P-RDL! For good BER, SWS switching prevails for both retries scenario (S-RRL and S-RDL), while for worse BER, the RRL scenario prevails for both switching methods (S-RRL and P-RRL). This is a very interesting finding, showing the particular strengths of the two adjustment methods tend to be complimentary (i.e. one relates poor BER while another relates to good BER). Of course, the most important fact to remember is that *the methods proposed here do in fact yield an improvement*. The precise method which is best in any particular scenario may differ.

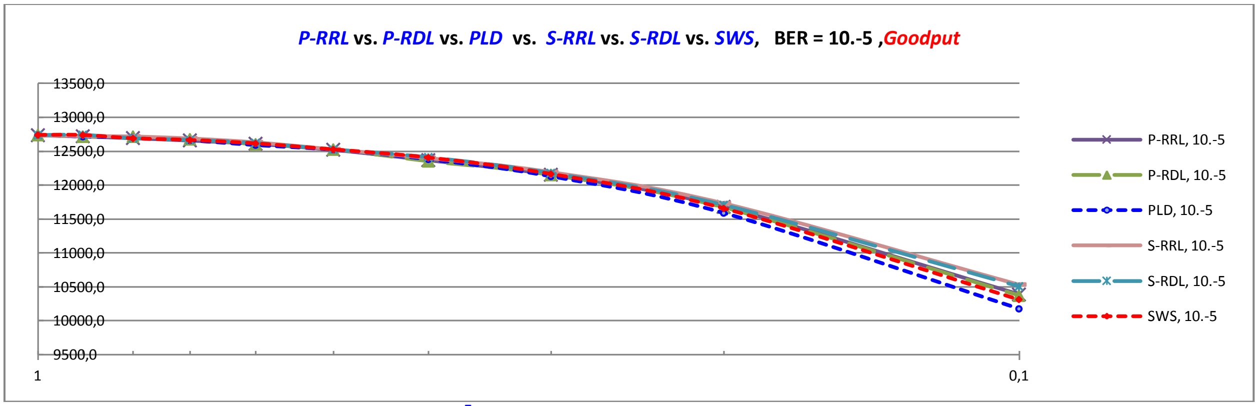

Fig. 2 – Link A getting worse, BER = $10.^{-5}$

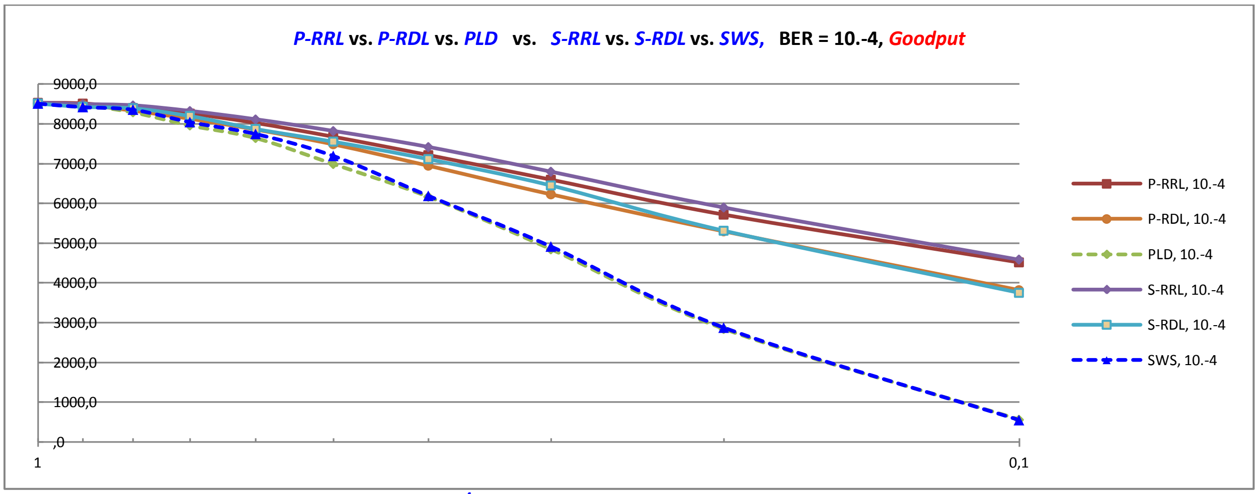

Fig. 3 – Link A getting worse, BER = $10.^{-4}$

So these two findings support the examiners' comment that different conditions will yield different results. This is one reason why the methods are clearly and openly explained in this thesis, and are very clearly outlined in the papers we have written about this: it is precisely so that other researchers can implement the method for their own systems, past, present and future (with the motivation that the system has been shown to work well on several test scenarios). We consider this to be a much more sensible idea than for one PhD candidate to have to test hundreds of possible alternative systems and scenarios!

In response to the examiner, we agree that it means the discussion needs to focus more on the sensitivities of the technique (i.e. what it is about the scenarios that leads them to exhibit a performance improvement with one of the techniques, and if so, which technique would perform best). So we have updated the thesis discussions in line with this.

Regarding the following experiments, since they continue with similar conditions to the original experiments (i.e. maintaining the original situation of Link A getting better), the assumption remains, that P-RDL is the best approach for this scenario and is hence adopted into the future experiments (unless otherwise stated in those experiments).

Chapter 6

A shortcoming of the proposed scheme is that there is no provision for raising the QAM level when link quality improves.

In this thesis, the down-adjustment was implemented with the knowledge that the previous (higher) QAM level is not sufficient to accommodate good transmission performance over the current channel conditions. For example, if the channel condition is bad at 64-QAM, there may be no other solution but to perform an adjustment down to 32-QAM. Once the condition becomes acceptable at the stepped down rate, then the system will continue to stay at that QAM level. Under the specific test implemented here, if the system then attempted to adjust back into 64-QAM (the previous erroneous channel), certainly it will fail again.

An up-adjustment scheme would certainly be possible if the system started with the lowest QAM level. One could imagine tracking the system performance, and stepping the QAM level up in response to a long, continuous, error-free transmission period. This would be good practice in an implementation, but not in simulation where VERY GOOD BER is problematic (since the simulation time can become extensive). It's much easier to operate a simulation system with many errors, since this triggers the error-response mechanisms under investigation much faster and more predictably.

Apart from this, the simulation was not building an implementation system – it was simply introducing and then evaluating a novel idea. The test did enough to validate that the idea can lead to improved performance. It is true that there are many practical issues to be overcome before this can be implemented in a product, however this research has clearly answered the question, "Is there any performance gain to be found through this technique?" For the first time, we have an answer. Now other researchers can answer the question "What is the best way to implement this idea in a practical system?" That question is considered to be outside the scope of the current thesis – although the candidate has included some hints and information that would be useful to those implementers.

Chapter 7

Discussions as expected.

Overall comments

1. It does perhaps contain too much background information, such as Chapter 2 on simulation tools and the first part of Chapter 4 on transport layer protocols.
   Regarding Chapter 2, a comparison table has been added to simplify the readers ability to surveying the conclusions observed about the various simulation tools. When this work started, OMNeT++ was not popular in the research literature, so effort was made to test and evaluate it. Although it is far more popular now, the evaluation may still be useful to some readers. Anyway, the chapter has been revised and clarified. Regarding Chapter 4, this chapter has also been re-written and summarised.

2. Simulation based rather than analytical performance evaluation, hence the validity of the results under different operation conditions is not clear.
   Yes, this is true. Although analytical equations have been given where possible, a large part of the performance evaluation rests on different simulations.

3. Incomplete definition of the proposed protocols, e.g. omitting details such as how timeouts are set and how multiple TCP packets are routed to avoid wasted slots.
   How timeouts are set has been addressed when answering question Chapter 3 – no. 1.
   How the TCP packets are routed has been addressed when answering question Chapter 5 – no. 1 & 3.
   All of this is now discussed within the revised thesis.

4. Not considering latency
   This has been addressed when answering question Chapter 4 – no. 3

5. Not investigating NAK
   This has been addressed when answering question Chapter 3 – no. 5

6. Lack of mechanism to raise the QAM level
   This has been addressed when answering question Chapter 6

____

Signature and Date