University of Technology, Sydney
School of Electrical Engineering
P.O. Box 123, Broadway 2007, N.S.W. Australia.

Thesis 1 Report: 'Congestion control in wide area TCP/IP using BONeS'

Matthew Gream
(90061060)
mgream@heckle.ee.uts.edu.au

Thesis Supervisor: Tamara Ginige

November 4, 1994

Total pages: 35

Contents

List of Figures

1 Introduction

The intention of Thesis 1 is to investigate and derive a specification for the work to be carried out in Thesis 2. At the initial stages of Thesis 1, a topic was worked out and it is the literature and research in the area of this topic that will be examined. Using the knowledge gained in this process, a system specification and a path towards resolving the particular problem associated with the topic can occur.

This report is structured to first go over the goals and objectives planned for Thesis 2. Next the literature and research that was carried out to examine issues surrounding the topic of investigation is presented. Following this, a work breakdown section outlines the work to be carried out in Thesis 2. Project management issues based on this work breakdown, such as time scheduling, task dependencies, contingency plans and milestones are devised and shown. Finally, a specification of the intended deliverables from Thesis 2 is given.

This document was written with the LaTeX package (Lamport, 1986; Maltby, 1992). A description of the Thesis topics can be found in (School of EE UTS, 1994).

2 The project and it's goals

2.1 Discussion

The specific project being carried out is "Congestion control in wide area TCP/IP using BONeS". This involves constructing and simulating wide area networks in different scenarios to investigate congestion.

The modelling and simulation is to be carried out using the 'BONeS Designer' package (Comdisco Systems Inc., 1993) in which components are constructed as modular 'blocks' and interconnected in a hierarchical fashion. In this approach, complex blocks can be constructed which present themselves as single black boxes with specified inputs and outputs, but internally consist of other interconnected blocks. At the bottom of each hierarchy are primitive blocks either supplied by the BONeS package (such as counters, queues, etc – a strong analogy can be drawn with logic circuits in digital systems).

Using BONeS, all the required modules will be constructed for use in 'building' diverse wide area networks for simulations. Completing the the construction of these BONeS components signifies an important milestone in Thesis 2, and it is after this point that the components can be strung together in different scenarios to examine congestion issues. It has been taken as an important aspect that the components be designed in such a way that they can be easily and effortlessly used in this manner.

Current simulations and simulators used to examine congestion control are either limited and do not model significant aspects of network activity, or lack scalability.

The simulations will attempt to look at several areas of congestion in wide area TCP conversations, over 'stateless' networks utilising a datagram oriented network layer protocol such as IP. The scenarios used for simulation will be wide area topologies that have been recognised as being representative cases of specific behaviour that has not yet been explored in this field.

The flexibility and completeness of the BONeS package and components will allow for insights not previously examined.

At the same time, while there have been many strategies devised to deal with congestion in these types of networks (including strategies for end systems and intermediate systems), there has been no 'back to back' comparison of proposed strategies in the complex scenarios envisaged above. This Thesis gives an excellent opportunity to not only examine the operation of these schemes in equal scenarios, but in scenarios that are complex and represent real network activity.

I'm personally interested in pursuing the relationship between dense multi-path networks with dynamic routing and load balancing. In this situation, transport connections are often rerouted through other nodes based on congestion at other nodes. This breaks the fundamental assumption that most congestion control methods are based on which is that the same RTT and hence the same path is used for the entire connection. The relationship between multi-path connections has not been examined yet, I hope to make some preliminary investigations into this area – that is the 'new' thing that will be investigated in this thesis.

2.2 Overall goals and objectives

The following 'top level' goals and objectives have been realised. The specific information they lack, such as for example the types of simulations to be performed, is filled in by the later breakdown of work.

- BONeS will be used to construct components that can be used to model and simulate TCP/IP based wide area networks.
  - The components will be modular and easily interconnectable so they can be used to construct various topologies and configurations.
  - The components will be constructed to be presentation quality.
  - It will be possible to generate and simulate using traffic that is representative of real wide area networks.
  - There will be support for various strategies that aim to deal with congestion – whether these strategies exist in end or intermediate systems.
  - It will be possible to record quantitative characteristics of each component. For example, an intermediate system with several incoming and outgoing links will allow examination of queue sizes, packets dropped, packets transmitted and so on for each output link.
  - All important aspects of an element will be modelled, but those that are not important will not be modelled. For example, it is not important to model the urgent data or large window details of TCP.
- The components constructed in BONeS will be used to model and simulate congestion in different wide area scenarios with a focus of examining deficiencies in current control strategies.
  - Models will be constructed to be intuitively understandable and presentable.

- A simple case will be used to verify the correctness of the model against results known from published literature.
- Simple scenarios will be used to provide a basic unified 'back to back' comparison of different strategies.
- Complex scenarios will be modelled and simulated to examine topologies and configurations previously unexplored. These models will represent actual real world networks or possible future situations.
- Traffic profiles and background traffic will be derived from actual wide area characteristics.
- Results from simulations will include important quantitative and qualitative details such as connection throughputs, delays, packet loss ratios along with issues specifically related to congestion such as intermediate system queue sizes, allocation of resources and so on.
- The results gained will be used in making an assessment of the simulations.

3 Research and Investigation

In order to understand the issues relating to the project, and hence be able to formulate a specification of the work to be performed, research was carried out. This included searching through physical and electronic ("online") resources.

3.1 Initial considerations

Before performing the investigation, I identified the areas I would need to examine. These areas are directly related to the goals and objectives stated above. Note that there was some iteration involved here, because as things progressed, I found myself re-establishing and altering some of the specific aspects of the projects goals and objectives.

The areas to be investigated were:

The TCP/IP protocol suite – This is the particular protocol suite being implemented and simulated even though much of the work is equally applicable to transport protocols in general. What is needed here is to examine the specifications of the protocol suite and issues surrounding it's implementation.

Congestion control issues in TCP/IP – Separate from the specification of the TCP/IP protocols are issues dealing with congestion control. It is required to examine the specific details of strategies used for congestion avoidance and control along with other work achieved, deficiencies and assumptions current work is based on and so on. The work in this area is the most critical. The implementation details will be used in the initial construction of the models, and the scenarios and results produced in the simulations will be driven by previous work and central issues.

TCP/IP in wide area environments – There are going to be characteristics of TCP/IP in wide area environments that need to be taken into account for the simulations. This includes models and observations of wide area TCP traffic characteristics and ways in which wide area networks are connected and utilised.

Flow control and congestion issues – Because congestion has been an issue for packet switched networks and other transport protocols, there is a need to examine flow control and congestion in a more generic sense. In doing this, a better understanding of needs and requirements for modelling and interpretation of results can be obtained.

Modelling with BONeS – As the software being used to carry out the modelling and simulations is BONeS, it's features, abilities and deficiencies need to be recognised. This work is going to be fundamentally important in constructing the initial components for modelling and issues relating to interconnection and simulation along with post processing for report generation.

Models and simulations – The statistical verification and assurance of simulation results is important. It would be of no use to carry out simulations without attempting to ensure the results are valid. The issues involved here are going to be those associated with monte-carlo simulations. This is largely a minor issue, as previous experience through course subjects dealing with modelling and statistics has already provided most of the knowledge here. However, it would be appropriate to examine literature that deals specifically with the case of network models and simulations.

3.2 Reviewed work

The approach I've taken here is to list all the literature surveyed along with a short description of what the item is about, and the details from it that are relevant to the work here. There are a few papers that could not be obtained, because they weren't available locally (UTS, Sydney Uni or UNSW). At the moment, I don't think they are of significant importance as the current set of research has uncovered what seems to be comprehensive information.

The order of presentation of this literature is insignificant.

3.2.1 The TCP/IP protocol suite

Postel (1981b) : This document (RFC-791) defines the Internet Protocol (IP)[1] and is the reference standard. It gives a contextual overview of the protocol and it's relation to other protocols before specifying datagram formats and operational requirements.

Postel (1981c) : The Transmission Control Protocol (TCP) is specified here in RFC-793. It contains an overview and philosophical design details before giving a functional specification of the protocol, including data formats and operational issues. In addition, Jacobson and Braden (1988) covers, inter alia, the extension that provides Selective Acknowledgements to the protocol. This 'SACK' option is an important component of

[1]Only recently, a new Internet Protocol was specified for mid term implementation and deployment, at this point in time the differences between the two don't warrant concern from our viewpoint. There is one feature of the new IP that could be relevant, this will be described later.

several congestion control strategies. Additionally, Clark (1982b), Clark (1982a), Postel (1983) and Sidhu and Blumer (1985) clarify certain implementation deficiencies and ambiguities.

Postel (1981a) : Here (RFC-792) the Internet Control Message Protocol (ICMP) is specified. This protocol acts to report errors and transmit control information at the network layer (IP). One such error message, the 'Source Quench', is reported back to a transmitter when one of it's packets is dropped by a gateway. Some TCP implementations use this in their congestion control strategy.

Braden (1989) : This RFC-1122 is an official specification in that it "amends, corrects, and supplements the primary protocol standards documents relating to hosts". In doing so, it specifies a congestion control scheme for TCP that was lacking in the initial specification and sets out which features are required or optional for a TCP implementation. Any TCP/IP implementation needs to conform to this specification.

Comer (1991) : This is a practical insight into the TCP/IP protocol suite. He describes, by way of the implementation he has written for the Xinu package, the structure of TCP/IP and all features of implementation including ARP, IP, TCP, UDP, ICMP and even RIP and SNMP. It is educational both in describing the working of the protocols themselves and in observing how they coalesce together in an actual implementation.

Stallings (1991) : An overview of the whole computer communications arena. It covers many of the generic issues and terminology along with an introduction to Queueing Analysis. Another general networking book is Spragins, Hammond and Pawlikowski (1992), which was used in a previous subject, Data Communications.

Stallings (1993) : A more formalised description of standards, including the TCP/IP protocol suite. It's essentially a reference publication that provides a summary and implementation details on major standards in 'half way' language.

3.2.2 Congestion avoidance and control in TCP/IP

Nagle (1984) : Nagle describes the phenomenon known as 'congestion collapse' and outlines two problems with TCP implementations. The first of these is due to too many small packets (this was before the 'silly-window' technique was developed) and the second is partially a result of the first : excessive congestion. He proposes that transmitters should reduce their transmit window size when a 'Source Quench' is received, and then return to normal after a certain number of ACKs have been received. Unfortunately, this is not a good solution and follow up papers have pointed this out.

Jacobson (1988) : Essentially a seminal work on congestion avoidance and control in TCP. He actually observes Nagle's conjectured 'congestion collapse' in operation and provides several new algorithms for TCP to counteract the problem. His first is the 'slow-start' algorithm that ensures transmit window sizes increase linearly from one after a timeout or at startup. Eventually at some specific window size a packet is lost due to congestion, this is implicitly detected by a timeout or reception of duplicate ACKs. The window then reduces multiplicatively and re-enters the slow-start phase. Other algorithms focus on better RTT estimates and fast retransmits. His basic philosophy is 'a conservation of packets', in that new packets should not enter the network at a rate faster than they are removed.

end2end-interest (1988+) : There is a considerable discussion spanning several years on the topic of congestion avoidance and control in the 'end2end mailing list', whose charter is to discuss issues relating to end to end protocols. Within these are modifications and changes to Jacobson's original congestion control algorithms.

Mankin and Ramakrishnan (1991) : The purpose of this paper is to "present our review of the congestion control approaches, as a way of encouraging new discussion and experimentation". It describes the terminology in use and performance goals quite extensively. Gateway congestion control measures are examined, which include 'Source Quench', packet drop policies, congestion indication bits[2] and Fair Queueing. In discussing end-system congestion control policies, some attention is given to existing problems, but only Jacobson's work and Jain's 'Congestion Indication' approach are tackled.

Wang and Crowcroft (1992) : An analysis and simulation of Jacobson's slow-start and congestion avoidance algorithms is shown where oscillatory patterns are visible. The DUAL algorithm is presented that provides for rapid adjustment when traffic changes occur and minimal oscillation once convergence has been reached. The three simulations performed are limited to a) one user over the path, b) two users over the path and b) two-way traffic.

Zheng Wang (??) : The authors separate the flow control window from the congestion control window (as the former is a relationship between the two end systems and the latter is a relationship between the end system and the network). The show a phenomenon known as 'pipe breakdown' that occurs with a single window, and use this as a a means of comparison with their dual window solution. They note that there needs to be a way of comparing the two schemes in a controlled environment.

Wang and Crowcroft (1991) : After giving an analysis of end-point control schemes the 'Tri-S' scheme is introduced. The implicit detection of a lost packet is used to locate the 'operating point' of the network (as with Jacobson's approach), the scheme then monitors the throughput gradient of the network in order to predict incipient congestion. They use "five carefully selected simple scenarios: 1). one user using one path, 2). two users sharing one path, 3). two users sharing one path partially, 4). one user joining in a steady flow, 5). one user leaving a steady flow." in their simulations.

Floyd (1994) : Floyd examines the use of Congestion Indicated, or 'Explicit Congestion Notification (ECN)', mechanisms in TCP/IP. The issues surrounding the implementation of this scheme, including addition of 'bits' into the IP header plus gateway response messages, is discussed. Simulated scenarios focus on the use of different gateway policies in parallel having bulk-data connections into a single congested gateway. As of writing this, Floyd has released a revised edition of the paper – I haven't had a chance to examine the differences yet.

[2]Unfortunately IPv6 was specified without a CI bit.

Brakmo, O'Malley and Peterson (??) : 'Vegas' is the most recent congestion control measure proposed. Three new techniques are employed, the first uses a more accurate RTT estimate involving local time stamps. Returned ACKs can be matched with time stamps to pre-empt retransmissions. Like Tri-S it also uses throughput gradients to detect incipient congestion using fixed thresholds to determine when to reduce the transmit window size. Their third change is to modify slow-start to occur only every other RTT, so as to get a better estimate of expected and actual throughput rates. Their simulations involve two simultaneous connections with and without background traffic (including two-way traffic), different TCP send-buffer sizes and multiple competing connections. Their real-world results also look impressive. They believe more work needs to be done on examining fairness.

Floyd and Jacobson (1993) : In stating that "Only the gateway has a unified view of the queueing behaviour over time ..." amongst several other statements, the authors believe that Random Early Detection (RED) gateways can detect incipient congestion by monitoring average queue sizes. When congestion occurs, packets are dropped or marked (a Congestion Indicated bit is assumed) with a certain probability that is dependent on the average queue size. They present previous work on congestion avoidance gateway strategies and then give algorithmic details for the operation of RED gateways. Simulations are performed with multiple sources feeding a single sink through a congested gateway with each source having a different delay-bandwidth product. Parameter sensitivity is also analysed and other simulations include looking at bursty traffic. Note that this paper works together with Floyd's (1994) ECN proposal.

Floyd (1991a) : This paper examines biases that exist with current congestion control schemes. This includes biases against connections with multiple congested gateways, against connections with longer round trip times and certain gateways against bursty traffic. Simulations involve a connection with bulk and bursty traffic across several gateways that have individual 'cross' traffic. There is quite a mathematical treatment with a focus on fairness issues. It is suggested that further work is needed on solving the problem of why connections through multiply congested gateways should receive unacceptably low throughput.

Floyd (1991b) : In this, the second part of Floyd (1991a), the simulations are enhanced with two-way traffic to generate the 'compressed ACK' phenomenon. The bias that against bursty traffic by 'drop tail' gateways is exacerbated by the addition of two-way traffic. This isn't a result of the two way traffic specifically, but a side effect of compressed ACKs.

Wang (1992) : Wang's thesis is a fairly hefty treatment of routing and congestion control. He proposes, inter alia, a new mechanism for 'traffic adjustment' to be used with congestion control schemes. The scheme is, in fact, that presented in (Wang and Crowcroft, 1991) as Tri-S and (Wang and Crowcroft, 1992) as DUAL. The simulations scenarios are still relatively simple and with limited topologies.

Floyd and Jacobson (1992) : This is an interesting paper describing the effects of the bursty traffic generated by window based flow control schemes. The problem in this case is that the periodic traffic fills gateway queues and starves other connections attempting to transmit at the same time (if their 'burst' only just follows that of the other). It is possible that some traffic may receive a substantially disproportionate share of bandwidth. The authors then examine the role of gateway congestion control policies in this activity and how specific action can eliminate these phase effects. The simulations used are again relatively simple topologies.

Mogul (1992) : Mogul sets up real world surveillance to examine the 'compressed ACK' phenomenon in operation. The result is that he finds that it does occur and is detectable. While most of the report is taken up with practical issues of the project, there are some important points in the description of how ACK compression is detected. An interesting conjecture that ACK compression may be correlated with segment losses and an analysis suggests that it may be so, but the evidence is not conclusive. Time stamps on returned ACKs are suggested as a counter-measure.

Huynh, Chang and Chou (1994) : The authors present a rate based congestion mechanism that they claim functions better than the existing window based schemes in TCP. Their simulations with one and two way traffic seem to support the assertion that the technique exhibits better stability and fairness. Their algorithm looks sensitive to fluctuating network rates.

3.2.3 Wide area TCP/IP

Paxson (1991) : Wide area TCP conversations are measured and examined over two one month periods. Traffic is examined in terms of it's geographical distribution, and more importantly, in terms of it's quantitative nature. The behaviour of several of the more popular application protocols (telnet, FTP, SMTP, X11, etc) is modelled by fitting the data to distributions. However, the qualitative measure used to assess the confidence of the modelled data in most cases reveals that their models is not good.

Paxson (1993) : Following on from Paxson (1991), the authors have made a 2 year analysis of wide area TCP conversations. While no attempt is made to try and model the traffic's behaviour, the paper provides a good analysis of real world conversations and issues involved.

Cáceres, Danzig, Jamin and Mitzel (1991) : Wide area TCP conversations are analysed and from the models of traffic are developed. Their results "... outline the necessary steps to describe and simulate a new conversation between two networks." Importantly they find that between 25% to 45% of packets are attributable to interactive conversations which is interesting considering that most current simulations focus on bulk data conversations. The paper gives an excellent treatment of conversation characteristics and a number of suggestions for those carrying out simulations.

3.2.4 Generic flow control and congestion issues

Jain and Ramakrishnan (1988) : This is an excellent overview paper on the field of congestion control. It compares flow control to congestion control, the design requirements for congestion avoidance and control, existing schemes, performance metrics, goals and miscellaneous related topics.

Tipper, Hammond, Sharma, Khetan, Blakrishnan and Menon (1994) : The focus of this paper is on a virtual circuit based network, but the authors investigate the effects of link failure and the subsequent effects of rerouted traffic. Treatment is mathematical to a large extent. An important observation is that there are significant localised congestion effects as traffic is rerouted. The network however uses call admission in which it determines an optimum path for the virtual circuit to take. This isn't applicable to stateless networks.

Comer and Yavatkar (1990) : The authors work from the premise that existing schemes rely on connection-oriented transport protocols to restrict traffic flows, when in fact connectionless traffic such as UDP may form a significant component of network traffic. Also, they note that newer transport protocols use rate based schemes and that feedback schemes exhibit control delay. Their rate control technique works within the network, and depends on nodes obtaining status information about all other links. A limiting factor is the fact that "for each destination, a node on the periphery computes the shortest path and the total capacity available on the path." These nodes also return rate reduction messages to the sources. Initial simulation results look promising.

Robinson, Friedman and Steenstrup (1990) : The BBN packet switch congestion control algorithm, in contrast to many other schemes, involves the integral use of intermediate gateways to compute flow information and related information then flood this through the network.

Zhang, Shenker and Clark (1991) : In setting up an environment and simulationing two way traffic, the authors show the existence of ACK compression and other dynamics that were either not present or minimal in the one way traffic case. Additionally there is a synchronisation phenomena occurring as well, this is actually fully explored in a later paper by Floyd on phase effects (Floyd and Jacobson, 1992).

Liebeherr and Akyildiz (1990) : Four different configurations are studied, these are i) no capacity constraints, ii) gateways with capacity constraints, iii) channels with capacity constraints, iv) channels and gateways with capacity constraints. The networks are represented using queueing models. The results display relationships between congestion and varying speed network components.

Ulusoy and Baray (1988) : This paper examines "a window based congestion control mechanism ... in an interconnected network". Their simulation identifies the relationship between delay and throughput for fixed and dynamic window based control. The conclusion clearly points out qualitative benefits of the dynamic control.

Nagle (1987) : Nagle shows how infinite queue/buffer sizes will not prevent congestion occur, and in fact exacerbate it. The basic problem is that with infinite queues, transmitters will 'timeout' and assume the packet has been lost, hence retransmitting it and injecting packets into the network faster than they are removed. In addition, packets have a 'time to live field' which means as they exit large queues, they will almost instantaneously be discarded as this field has counted down to zero while waiting in the queue. Half of the paper is devoted to issues relating to fairness.

Chiu and Jain (1989) : Most of this paper is given to a mathematical treatment of the AIMD scheme and it's application to congestion control. In this treatment it looks at network performance as a function of load, using both response time and throughput. The degradation in response time occurs as network queues build up.

Jain (1990) : This is a general paper providing an overview of the issues in congestion control. Jain first examines the popular myths that exist about congestion control, especially those that assume the problems will be solved by faster links, processors or buffer space. The currently known schemes are classified according to whether they involve resource creation or demand reduction, and then the goals of congestion control policies are articulated. Finally, three schemes are specifically described. Jain notes some areas for further research, one of which deals with path splitting.

Jain (1986) : The scheme described in this paper is the well referenced 'CUTE' scheme. It is fairly simplistic in that the window size is increased 'parabolically' until a timeout occurs, indicating a packet lost to congestion, at which time the window size is reset to an initial value of one. It's clear that this scheme has oscillatory problems. His simulation involves a simple point to point link occupied by three connections.

Agrawala and Sanghi (1992) : This paper focusses on examining congestion control in the face of cross traffic, where most prior studies have assumed this not to be present. The treatment is analytical and the simulation relatively simplistic.

3.2.5 Modelling with BONeS

Comdisco Systems Inc. (1993) : The BONeS user manual contains tutorials and reference information. Chapters 2 and 3 have been examined, but only limited examination of the rest of the manual has so far occurred – this will be rectified soon though.

Munro-Smith and Farhangian (1993) : An overview of the construction and simulation of a wireless LAN system. The key important feature of this report is that it's able to give some insights into what BONeS is capable of, along with showing the possibilities with post processed outputs of simulations.

Thompson, Moorhead and Smith (1992) : This paper describes the considerations and then the models that were constructed to simulation a combination of FDDI and CSMA/CD networks. Detailed diagrams are given of models, including traffic sources, and generated statistics. There is an important discussion about using BONeS to construct models and run simulations, including a paragraph from the conclusion:

> The analysis of the ASRM network was simplified by using the commercial software BONeS. The software allows the user to graphically build a network. The ASRM simulation was built using several components from the BONeS library. BONeS also allows the user to build his [sic] own modules. A very detailed simulation can be accomplished with BONeS. However, this also means that building a simulation can be a complex task and the actual time to do the simulations can be very long.

Shanmugan, LaRue, Klomp, McKinley, Minden and Frost (1988) : This paper is a guide to BONeS, its 'network modelling philosophy' and many other issues. It serves as a good introduction and overview to BONeS in terms of it's background, philosophy and capabilities.

Kavi, Frost and Shanmugan (1991) : A case study of the construction and simulation of an LLC protocol using FDDI. Details of how modules were put together and simulated are given, mainly focussing on a "methodology of implementing LAN protocols so that they can become part of a protocol library".

3.2.6 Models and simulations

Miller, Freund and Johnson (1990) : Used as the text for an earlier subject, Engineering Statistics, this is good for describing sampling distributions and the application of them in making inferences about samples. It is expected that some simulations will need to be carried out multiple times and results analysed to ensure confidence in the mean and variance of such results.

Hamburg (1970) : This is an introductory text to statistics, but it's major focus is on quantitative techniques for decision making and analysis. The important information in here is the validation and analysis of statistical results.

3.3 Examination of other items

As the BONeS software is the primary element in the operation of this project, it was deemed essential to gain some initial experience with it. What I've done so far is:

- Examined the Library guide to see what components are available and how they operate.
- Examined the 'Wide Area' and 'Campus Network' examples.
- Gone through the tutorial using the 'Stop & Wait Example'.

I was planning to construct my own Stop and Wait example from scratch and then run it, but this will have to be delayed at the present moment because of time constraints.

4 Work to be carried out

In this section, a discussion is first given to the actual elements that are going to be in the subsequent work specification. This specification outlines, to as much detail as is currently available, the tasks that are going to be carried out during the course of Thesis 2. It was developed in a top down fashion, a typical systems engineering approach – cf. (Aslaksen and Belcher, 1992).

4.1 Issues to look at

From examination of the literature on the topic of congestion avoidance and control, it becomes clear that a number of 'methods' have been devised and proposed. All of these have been either given a mathematical treatment or examination in a limited simulation scenario. These scenarios have often not taken into consideration the more realistic and complex effects of 'real' wide area networks. In fact, some researchers have shown up deficiencies in this area.

What I propose to do, firstly, is to provide an examination of the current congestion control methods that have been proposed in an equivalent environment. The purpose of this is to provide a comparison in terms of the recognised 'important' issues such as fairness, optimality, stability and so on. This will serve as a 'survey' of current methods.

The other problem with current schemes is that they use a Round Trip Time (RTT) from the conversation under management as an important input to their adjustment algorithms. From the research I have performed, there has been limited (vritually none, only hints of it in suggested future work) examination of the effects of multiple paths. It is well known that in complex interconnected wide area networks, such as the higher level networks in the US section of the Internet, that packets from a single conversation may take different paths through the network, experiencing different levels of congestion and delays.

It appears to me that in this case, where intranetwork switching occurs and when end systems are not informed, that the effects of congestion can be exacerbated. I am planning to construct a model and simulation to examine exactly what happens in this situation, to determine whether or not the effects of 'multiple paths' are significant and to see whether or not congestion based path switching could increase the temporal level of congestion within the network. In addition, mobile-IP proponents suggest that intranetwork 'redirect' messages should be propagated to end systems so that they can 'reseed' their RTT estimates, so it would be good to look at the problems surrounding this.

Note that the 'multiple paths' I am referring to are where packets from the one conversation take different paths during the conversation, not where the 'entire' path itself is shunted or altered – as this latter change can be modelled simply as the entry and exit of conversations.

To construct these models and simulations, I will use the BONeS package which allows components and models to be constructed and simulated easily and seamlessly.

Therefore, the important aspects of this thesis are the use of BONeS in constructing and simulating the scenarios and the models. The simulations carried out will in the first degree provide a benchmark comparison between existing schemes in a realistic manner that has not been investigated – ie. with 'real' background traffic. Secondly, they will hopefully provide some insights into an area that has not yet been given attention.

I hope that it will be possible to use the work in this thesis as a basis for constructing a publication quality article. I'm certain that this work will be of use to others in this field.

4.2 Elements in the work breakdown

The work required for Thesis 2 can be partitioned into three top level tasks. These are:

- Constructing the BONeS components and data structures.
- Putting the components together to create and run simulations.
- Interpreting and presenting the results of the simulations.

In my case, I intend to incorporate the latter task into the two former. For the former two, I've determined an initial specification for the work that needs to be done in a qualitative manner. This is articulated in the following two subsections.

4.2.1 Constructing BONeS components

I've identified that the following top level components need to be constructed. I'd like to go further down the hierarchy in specifying these components, but haven't had the time to do so. What I have done is outline the top level, and then a sketch of the next level down.

These components have been specified because I thought it was important to make them as orthogonal as possible. That means having the three fundamental network components: End Systems, Intermediate Systems and Links. These are supported by a LAN unit responsible for modelling the aggregation of traffic at a local area level. To support the requirements for examining traffic, there are two other additions. The first is a traffic event generator that is responsible for instructing a TCP entity to start, stop and maintain (ie. supply data to) conversations. The second is a traffic generator that transmits IP datagrams according to 'real' measured wide area traffic characteristics (to provide background noise).

The operation of the TCP stack is derived from the TCP specifications and encapsulates the important functional requirements relevant to the modelling that I will perform.

End System (ES)

Description:

> The ES will model a full duplex TCP/IP stack with the functionality required to carry out TCP conversations. It will not contain all elements of the TCP protocol, only those required to carry out the core functionality of TCP and support congestion control mechanisms. The control for the ES will come from the ESEVENT component which supplies commands and traffic. In operation, the TCP component will package and transmit TCP-PACKETs to a specific destination. The IP component is responsible for verifying and deconstructing received packets and constructing packets for transmission. The ES will also 'record' important information such as throughput and so on.

Derivatives:

> BSD43-ES – BSD4.3 (Jacobson, 1988) congestion control.
> VEGAS-ES – Brakmo et al.'s (??) congestion control.
> DUAL-ES – Wang and Crowcroft's (1992) dual window congestion control.
> RBA-ES – Rate Based Acknowledgement congestion control.
> TRIS-ES – Wang and Crowcroft's (1991) Tri-S congestion control.

Operation:

IP accepts IP-PACKET at input.
IP verifies IP-PACKET and strips off DATA portion.
IP passes DATA and EXTRA to TCP.
TCP performs TCP processing.
IP accepts TCP-PACKET and ADDRESS from TCP.
IP constructs IP-PACKET.
IP sends IP-PACKET to output.

Parameters:

ADDRESS – A unique address for this TCP end-point.

Inputs (number):

IP-PACKET (1) – Input packets.

Outputs (number):

IP-PACKET (1) – Output packets.
TCP-COMMAND (1) – Traffic characteristics and control.

Data structures:

EXTRA – Contains information from the IP-PACKET such as RTT, CE-BIT and so on.

Internal construction:

Figure 1 shows an expected realisation of the ES at the top level.

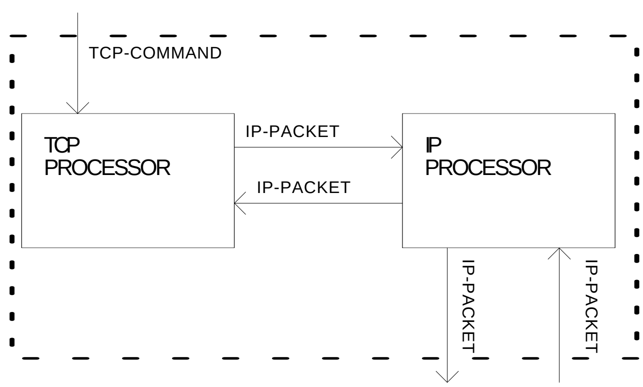

Figure 1: Top level representation of the ES component

Link (LINK)

Description:

The LINK models a full duplex data communications link. At each input it takes IP-PACKETs and queues them for a time corresponding to the bandwidth and delay of the link. Upon leaving the queue they emerge at the output. In addition, there is a control output that informs the connected component that the LINK is either able to accept another IP-PACKET, or is 'DOWN'. To simulate the actions of a link going 'UP' and 'DOWN', events are read from an even file. Note that there is a 'clear to send' indication to assert input flow control.

Derivatives:

none.

Operation:

Accept IP-PACKET at input.
Output IP-PACKET after time $\frac{IP-PACKET.LENGTH}{BANDWIDTH} \times DELAY$.
Output LINK-CTS after time $\frac{IP-PACKET.LENGTH}{BANDWIDTH}$.

Read EVENT from EVENT-FILE.
At time EVENT-ENTRY.START, set LINK-STATE to EVENT-ENTRY.STATE.
Output LINK-STATE on LINK-STATUS.

Parameters:

BANDWIDTH – Bandwidth in bits per second of the LINK.
DELAY – Propagation delay of the LINK.
EVENT-FILE – Filename to read EVENT-ENTRY information from.

Inputs (number):

IP-PACKET (2) – Input packets, rate controlled by LINK-CTS.

Outputs (number):

IP-PACKET (2) – Output packets to LAN or IS.
LINK-STATUS (2) – Notify of LINK status.

Data structures:

LINK-STATUS = Set of LINK-DOWN, LINK-UP, LINK-CTS.
EVENT-ENTRY = START time, STATE of link.

Internal construction:

Figure 2 shows an expected realisation of the LINK at the top level.

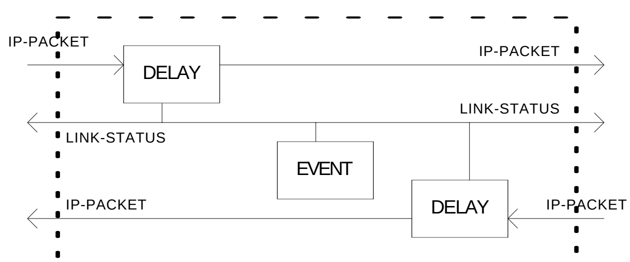

Figure 2: Top level representation of the LINK component

Intermediate Systems (IS)

Description:

The IS is the primary switching element in the network. It's inputs and outputs are IP-PACKETs to and from LINKs. For each received IP-PACKET, it will route it to an appropriate output interface based on an internal routing table (fixed at setup). This routing table will be subject to local conditions, such as congestion. The congestion policy for the IS resides in the INTERFACE component, it will be based upon an indicator in the TCP-PACKET that specifies what policy is in effect.

Derivatives:

IS-DROPTAIL – Model the 'random drop' style of packet discard.
IS-RANDOMDROP – Model the 'random tail' style of packet discard.
IS-RED – Model Floyd and Jacobson's (1993) RED gateway.

Operation:

Accept IP-PACKET from LINK.
Place in output INTERFACE according to next hop from routing table.
Discard IP-PACKET if output buffer is full.
Carry out congestion control in INTERFACE.
Update routing table based on 'DOWN' LINKs or congested INTERFACEs.

Parameters:

QUEUE-SIZE – Each INTERFACE can have a specified QUEUE-SIZE.
OVERHEAD – Each packet route can be given a fixed time overhead.

Inputs (number):

IP-PACKET (1...4) – From LINKs.

Outputs (number):

IP-PACKET (1...4) – To LINKs.

Data structures:

none.

Internal construction:

Figure 3 shows an expected realisation of the IS at the top level.

ES Event Generator (ESEVENT)

Description:

In order to provide control over the operation of the ES, the ESEVENT provides commands and traffic profiles. These commands include starting and concluding conversations, which maybe either time or volume scheduled. The ESEVENT will also generate data commands for the transport of data by the TCP in the ES. EVENTs are read from an EVENT-FILE which specifies the time for the EVENT to occur, and what is to occur.

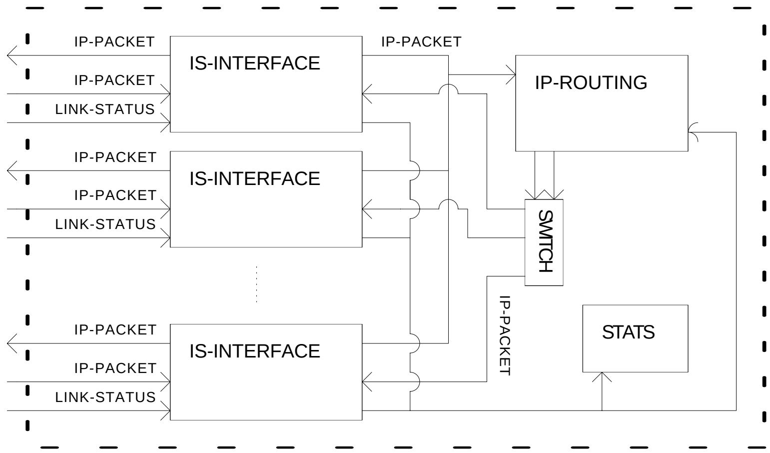

Figure 3: Top level representation of the IS component

Derivatives:

none.

Operation:

Read EVENT from EVENT.FILE.
At EVENT.TIME, send EVENT.COMMAND.
If EVENT.COMMAND is DATA, then until EVENT.END or EVENT.VOLUME, send data commands using traffic EVENT.TRAFFICMODEL.

Parameters:

EVENT-FILE – where to read EVENTs from.

Inputs (number):

none.

Outputs (number):

TCP-COMMAND – Control to an ES.

Data structures:

EVENT – TIME of event, END time of event, VOLUME to transfer, TRAFFICMODEL to use and COMMAND to send.

Internal construction:

Figure 4 shows an expected realisation of the ESEVENT at the top level.

LAN multiplex/demultiplexer (LAN-UNIT)

Description:

The LAN unit is responsible for multiplexing and demultiplexing IP-PACKETs from ES to a single IS via a LINK. This LINK can be used to model a high speed LAN environment with unconstrained bandwidth. For simplicity of the model, it's demultiplexing

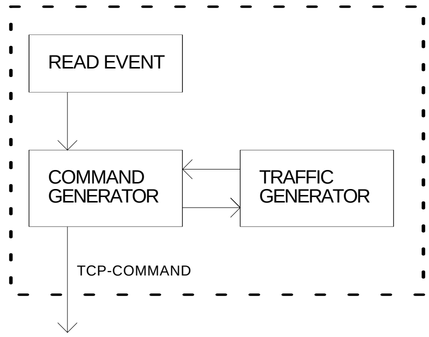

Figure 4: Top level representation of the ESEVENT component

will will involve broadcasting the IP-PACKET to all connected ES. The ES will filter input IP-PACKETs to discard those not addressed to it.

Derivatives:

none.

Operation:

Accept IP-PACKET from ES.
Transmit IP-PACKET on LINK.
Accept IP-PACKET from LINK.
Transmit IP-PACKET to all connected ES.

Parameters:

none.

Inputs (number):

IP-PACKET (1...8) – Provides input from ESs.
IP-PACKET (1) – Packets supplied via LINK from IS.

Outputs (number):

IP-PACKET (1...8) – Supplies packets to ESs.
IP-PACKET (1) – Supplies packets to LINK (thence to IS).

Data structures:

none.

Internal construction:

Figure 5 shows an expected realisation of the LAN-UNIT at the top level.

Background traffic Generator (TRAFFIC)

Description:

In order to provide realistic traffic on the simulated network, TRAFFIC will generate IP-PACKETs with distribution based on that obtained from wide area traffic profiles.

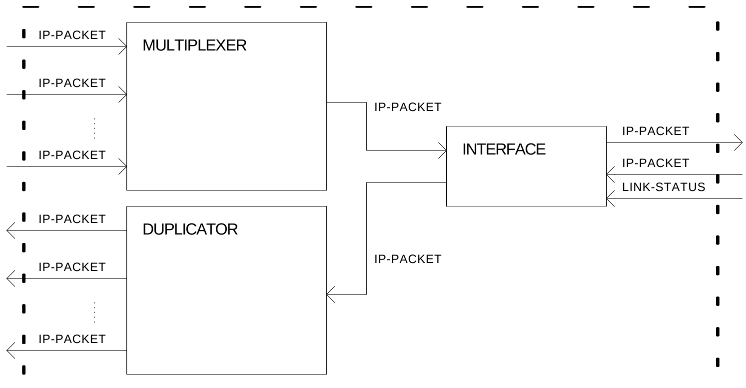

Figure 5: Top level representation of the LAN-UNIT component

These will be either transmitted to another TRAFFIC, or to an ES where they will be discarded. Control will come from a file specifying a time, duration and type of traffic to be generated with a corresponding destination address.

Derivatives:

none.

Operation:

Read EVENT from EVENT-FILE.
Construct IP-PACKETs at rate and size specified by EVENT.PROFILE to destination EVENT.ADDRESS.
Sink any input IP-PACKETs.

Parameters:

ADDRESS – A unique address for this unit.
EVENT-FILE – File to read event information from.

Inputs (number):

IP-PACKET (1) – Packets from LAN-UNIT.

Outputs (number):

IP-PACKET (1) – Packets to LAN-UNIT.

Data structures:

EVENT-ENTRY – START of event, LENGTH of event and PROFILE of traffic.

Internal construction:

Figure 6 shows an expected realisation of the TRAFFIC at the top level.

Notes on the design

It should be noted that as of this time, routing tables will be static. This means they will need to be 'programmed' for each model and simulation. This isn't a bad thing since the the time

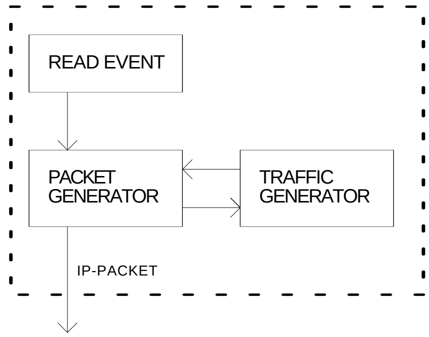

Figure 6: Top level representation of the TRAFFIC component

to do this this will outweigh the time that would have been spent building routing into the system. However, any local conditions such as a LINK failure will invalidate a specific route. It is planned that routing is global (for modelling and simulation purposes, not intended to be representative of an actual centralised routing mechanism). When a LINK fails, then the connected IS is able to invalidate it's routes using that LINK. At the same time, it will be struck out in a global routing table, and hence packets may be switched to a different path earlier in the chain. This does represent actual network operation, in which failed LINKs are reflected in routing updates which progressively invalidate that path.

An issue to be considered here is the progressive invalidation which may indeed cause transient congestion as packets are following a non-optimal path.

4.2.2 BONeS data structures

The data structures are the elements that flow between the created components. These data structures were derived by first considering what was required from the specification of the protocols (for example, sequence numbers and window information is required) but leaving out elements that are superfluous (such as options for 'large' windows and some TCP flags, etc). These were then supplemented by items specifically needed for the modelling and simulation, such as adding time stamps to record when the packet was generated and transmitted, and carrying a congestion experienced bit that is used by some congestion control schemes, but can is redundant when used with those that don't support it.

System wide, there is only one 'type' of packet – this is an IP-PACKET, but internally it carries a either a TCP-PACKET or a ROUTE-PACKET. The latter has not been defined yet. It may seem as though separate packets aren't required, but I have done so in order to logically separate the different items. Note that as stated above, the dynamic routing will *not* be constructed unless construction of the BONeS components finishes well ahead of schedule.

The overhead of this partitioning is not considerable, it requires minimal extra work and the advantages in terms of possible expandability are important. When it was later decided to add a 'traffic' unit, this partitioning looked even more appropriate.

The main 'system wide' data structures are:

TCP-PACKET

The TCP packet does not need addressing information, as this is conveyed by the IP layer and there is a one to one correspondence with TCP and IP entities in this modelling environment. What is first required are the items specifically related to a TCP conversation (from the TCP specifications):

- SEQUENCE NUMBER – sequence number of this packet
- ACKNOWLEDGEMENT NUMBER – reverse direction acknowledgement piggyback.
- flags: ACK, RST, SYN, FIN – for opening, maintaining and closing of connection.
- WINDOW – current window size.
- DATA OFFSET – size of data in packet (but note that no data is actually carried).

The parameters required for the simulation, and dealing with timing values and so on are carried with the IP-PACKET. This is done because these values are applicable to other items 'carried within' an IP-PACKET. The only additional item carried within the TCP-PACKET is a field to indicate what type of end to end congestion control scheme is in operation.

- CONGESTION-SCHEME – the congestion control sceheme in use for this packet.

There is no support for extended TCP features such as 'big windows' and urgent data – these are not required for the simulation environment being operated in.

IP-PACKET

The IP packet is a general network layer packet that carries an upper layer protocol packet (ie TCP) and addressing and related information. The fields that are required for the functional requirement of a network layer protocol are:

- SOURCE-ADDRESS – originator of this packet.
- DESTINATION-ADDRESS – recipient of this packet.
- CONTENT-IDENTIFIER – set of what is contained in the packet (ie. TCP, TRAFFIC, ROUTE or ICMP).
- CONTENT-LENGTH – length of encapsulated packet.
- CONTENT-DATA – the actual encapsulated packet.
- TIME-TO-LIVE – a hop counter.

Note that there is no need for features such as fragmentation, checksums and extended options, these are not required in this environment.

In addition to the functional requirements for operation of the protocol, there are items that are carried and used for simulation purposes:

- TIME-CREATED – when the packet was created and released into the network.
- CONGESTION-EXPERIENCED – single bit indicating that the packet received congestion in the network.
- PRIORITY – indication that this packet should be given priority in the network.

- EXTRA-TIMESTAMP – an extra timestamp able to be used by congestion control schemes.

Even though only some congestion schemes use congestion experienced bits (analogous to ATM's FECN bit), it has to be included here so that different schemes can inter-operate but not have differing packets.

ROUTE-PACKET

This packet isn't and won't be defined unless there is extra time available.

ICMP-PACKET

The ICMP packet exists only to provide reverse congestion indication to a transmitter. It also allows for an IS to indicate that a particular path is not reachable, and hence allow modifications to routing tables to be back propagated.

It's will contain:

- ICMP-TYPE – a set either being REDIRECT or SOURCE-QUENCH.
- SOURCE-ADDRESS – the source address of the packet that generated this message.
- DESTINATION-ADDRESS – the destination address of the packet that generated this message.

I have not included those that are *internal* to a top level module, mostly because they are still at a stage where they may be revised. This also includes the TCP-COMMAND data structure which carries information from an ESEVENT component to an ES component.

4.2.3 Models and Simulations

At this point in time I have defined three main topologies to perform simulations on, but I propose to expand this to five or more simulations by the time Thesis 2 is complete. Each topology and simulation has been constructed to examine a specific need.

Verification

Topology:

There will be a single ES transmitting across an IS and two LINKs. A representative diagram is shown in Figure 7.

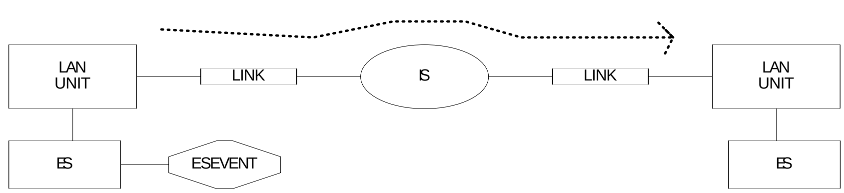

Figure 7: Verification simulation outline

Objective:

To ensure that the ES, IS and LINK are operating according to specification. That is, the LINK correctly models a communications link, the IS is able to accept and switch packets and the ES is able to maintain a TCP connection across these components. In addition, it will be possible to correlate these results with simulations performed by previous researchers.

Simulation parameters:

LINK – The DELAY and BANDWIDTH are arbitrary.

ES – Each of the different ES types will be used.

IS – Each of the different IS types will be used.

ESEVENT – The traffic supplied will be bulk data.

Simulation description:

The following 'variables' will be monitored:

LINK – THROUGHPUT.
IS – QUEUE-SIZE, PACKETS-DROPPED, INPUT-RATE, OUTPUT-RATE.
ES – THROUGHPUT, RETRANSMISSIONS.

The TCP conversation will start at time zero and continue on for a specified number of seconds.

Comparison

Topology:

There will be three LANs interconnected via a wide area connection having a single IS. Each of these LANs will have two ES in them. The setup is shown in Figure 8. The conversations are marked as being *A*, *B* or *C*.

Objective:

Each of the different congestion control schemes is to be tested in a controlled environment with equivalent parameters. The idea is to provide a 'back to back' comparison of congestion control schemes in terms of quantitative measures such as 'fairness', 'throughput' and 'delay'.

The topology is designed so that the conversation being examined *(A)* will experience congestion with another conversation *(B)*, and in addition will be subject to a conversation in the opposite direction *(C)* designed to introduce the effects of 'ACK-compression'.

There will also be background traffic introduced across the IS, but the details of this are yet to be worked out.

Simulation parameters:

LINK – The DELAY of *L1*, *L2* and *L3* will be equal and for a second simulation, *L2* will be twice that of *L1*.

ES – A simulation will be run to compare each type of ES with itself and each other type of ES. This will mean that the ESs associated with *A (E1, E2)* will be equal to the first type, and *B (E3, E4)* will be equal to the second type. The *C* conversation will remain constant.

IS – The wide area IS used will be a RANDOM-DROP.

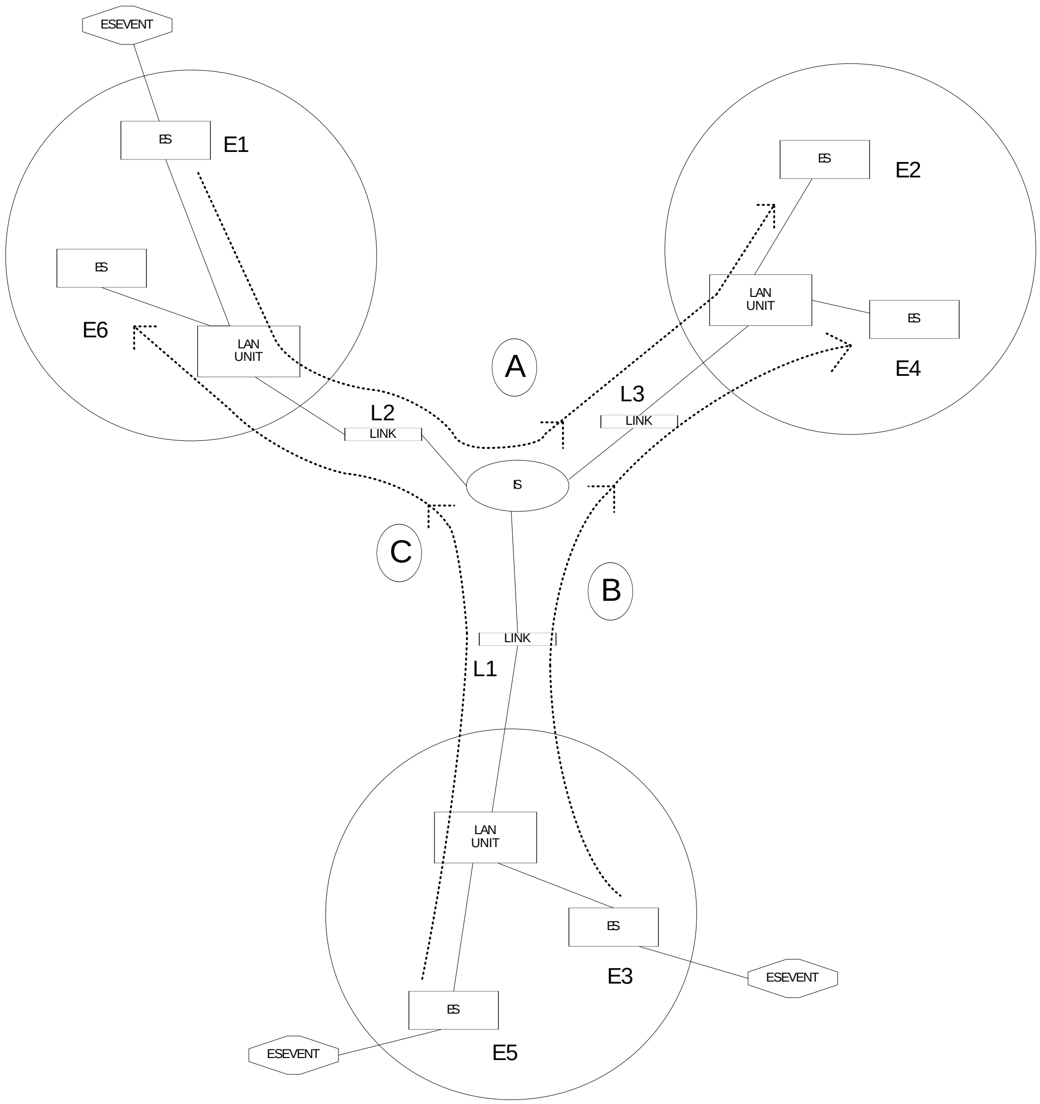

Figure 8: Comparison simulation outline

ESEVENT – The traffic supplied will be bulk data, and then interactive will be examined.

Simulation description:

The following will be monitored:

A, B – THROUGHPUT, RTT, EFFICIENCY.

IS – QUEUE-SIZE for A, B and C.

The conversation C will be started at time zero, then A after C has reached steady state. When A has reached steady state, then B will be restarted. After B has reached steady state, it will be removed. The result is an indication of both transient and steady state effects.

Dynamic networks

Topology:

The configured network has 4 LANs and 4 ISs. These are interconnected by LINKs of differing delay bandwidth products. There are two ESs that will carry between them a single conversation. There are multiple traffic sources that will be responsible for generating wide area traffic for 'queue loading'. See Figure 9.

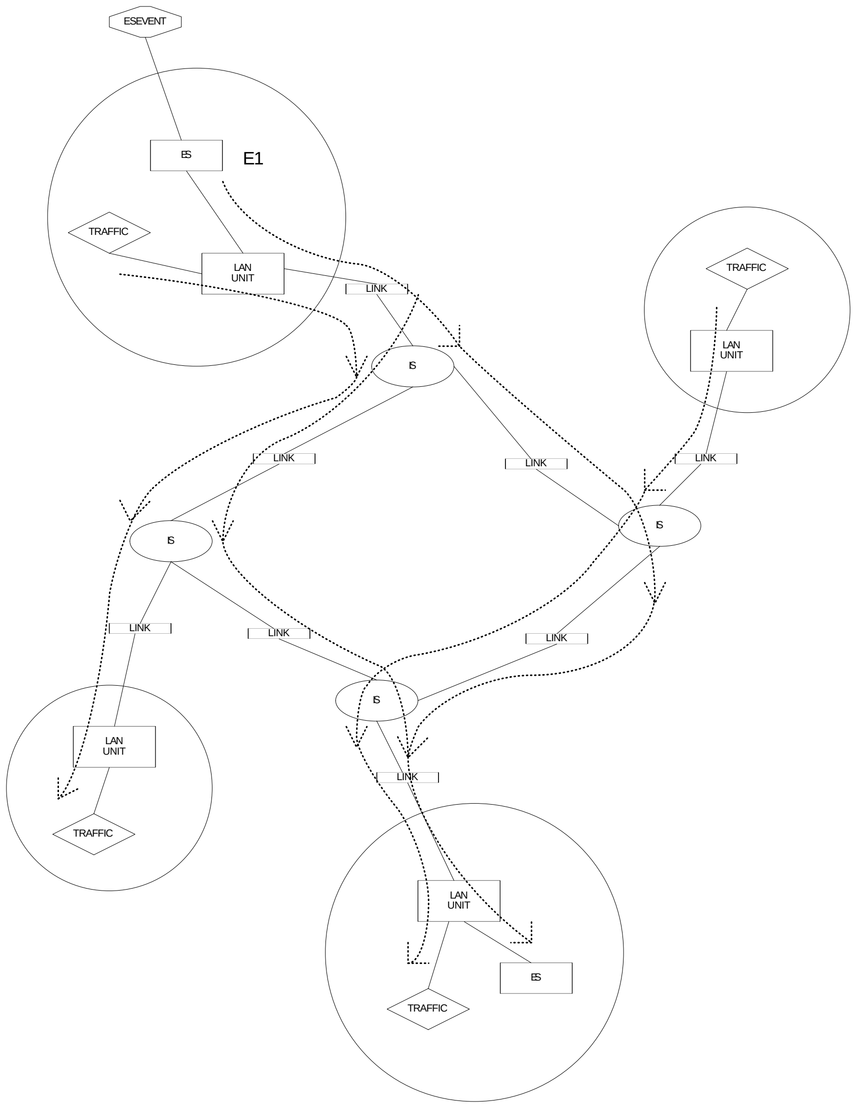

Figure 9: Dynamic network simulation outline

Objective:

The purpose is to examine what occurs when packets from a single conversations are spread across multiple paths with differing rates and RTTs. The ES *E1* will provide the conversation through the ISs that can take either of two paths. In addition to this conversation, wide area traffic will be generated on two of the ISs. The purpose of this traffic is to provide an environment where the conversation will experience different queue lengths and hence cause packets to be shunted between the two alternate paths.

Simulation parameters:

LINK – Each LINK will have a different delay bandwidth product. The 'outer' LINKs, those connecting IS to LANs, will have large enough capacity so as not to exert a considerable effect.

ES – The ES *E1* will begin it's conversation of bulk data at a short time after the simulation has begun.

TRAFFIC – Generated traffic will begin when the simulation starts. It will need to be 'bursty' in nature so as to cause the conversation to be switched between two different paths at the first IS.

Simulation description:

The following will be monitored:

E1 – THROUGHPUT, RTT, TRANSMIT.

The *E1* conversation will start at a short time after the simulation begins. At the start of the simulation the TRAFFIC generators will begin.

4.3 Work specification

The following tasks are required, with the estimated time indicated alongside. These times also include typing up the results for the Thesis 2 report and are specified in terms of days.

In deriving the times, I've used my experience so far with BONeS, the complexity required inherent in the particular task and the knowledge of my own strengths and weaknesses. Prior to working this out, I had hoped that it would take me about about a half to two thirds of the semester to construct the model, it looks as though it will take about that time.

The architecture of BONeS, that is the four main phases of data construction; block construction; simulation construction and finally post processing, already suggest a way to partition the work. This means that there is no point considering the aspects of constructing simulations and post-processed results until the data structures and 'blocks' are first finished.

The list of tasks along with expected length of time is shown in Figure 10. I have assigned August 1 as being the start of the work, but in reality I will most likely begin before that date – therefore it is a 'worst case' start time. The times also reflect weekends where no work is done. It will most likely be the case that work will be done on weekends.

The main milestone in the work is the completion of construction of the BONeS components.

4.4 Required resources

Outside of normal academic requirements, the only resource required for Thesis 2 is access to the BONeS software. Electronic access has already been given to the *mozart* machine, and physical access has been given by way of Room 2215 and *CSE* suns therein. The manuals for BONeS have also been made accessible in this room.

ID	Name	Duration	Start	Finish
1	Thesis 2	13.8w	1 Aug	3 Nov
2	Start Thesis 2	0d	1 Aug	1 Aug
3	Construct BONeS components	8w	1 Aug	23 Sep
4	Enter data structures	1d	1 Aug	1 Aug
5	Construct LINK	2d	2 Aug	3 Aug
6	Construct ES	4w	4 Aug	31 Aug
7	Initial TCP stack	10d	4 Aug	17 Aug
8	Derived TCP stacks	5d	18 Aug	24 Aug
9	Additional elements	5d	25 Aug	31 Aug
10	Construct ES-EVENT	5d	8 Sep	14 Sep
11	Construct IS	5d	1 Sep	7 Sep
12	Construct LAN-UNIT	4d	15 Sep	20 Sep
13	Construct TRAFFIC	3d	21 Sep	23 Sep
14	Complete BONeS components	0d	23 Sep	23 Sep
15	Verification of BONeS components	5d	26 Sep	30 Sep
16	Modelling and Simulation with BONeS	4.8w	3 Oct	3 Nov
17	Comparison scenario	4d	3 Oct	6 Oct
18	Dynamic path scenario	10d	7 Oct	20 Oct
19	Additional scenarios	10d	21 Oct	3 Nov
20	Complete Thesis 2	0d	3 Nov	3 Nov

Figure 10: Tasks to be completed during Thesis 2

5 Project Management

In the previous section, a breakdown of the work involved for Thesis 2 was given. This work must be performed in a finite time, therefore it is essential that it be scheduled to fit into these time constraints. Considering that only one person is working on this project, listing task dependencies might seem superfluous. In fact this is a good idea since there is the chance an unforeseen problem may arise while completing a task. Rather than dwell on the problem ('the law of diminishing returns') it may be better to step onto another task and put the problematic one to the side until it's been given clearer thought.

5.1 Gantt chart

The Gantt chart shows the scheduling relationship between tasks as a function of time. Most of the activities are carried out in series, although there is some parallelism possible with the incremental development of the Thesis 2 report. The Gantt chart is shown in Figure 11 and Figure 12

5.2 Network diagram

The Network diagram demonstrates the dependencies between each of the tasks, which can't be seen on the Gantt chart from the previous section. It's important to see that many of the initial tasks in constructing the BONeS model can operate in parallel *if* they need to. This was just previously discussed and is expanded on in the contingency plans.

The Network diagram is shown in Figure 13 along with the legend in Figure 14. It's clear

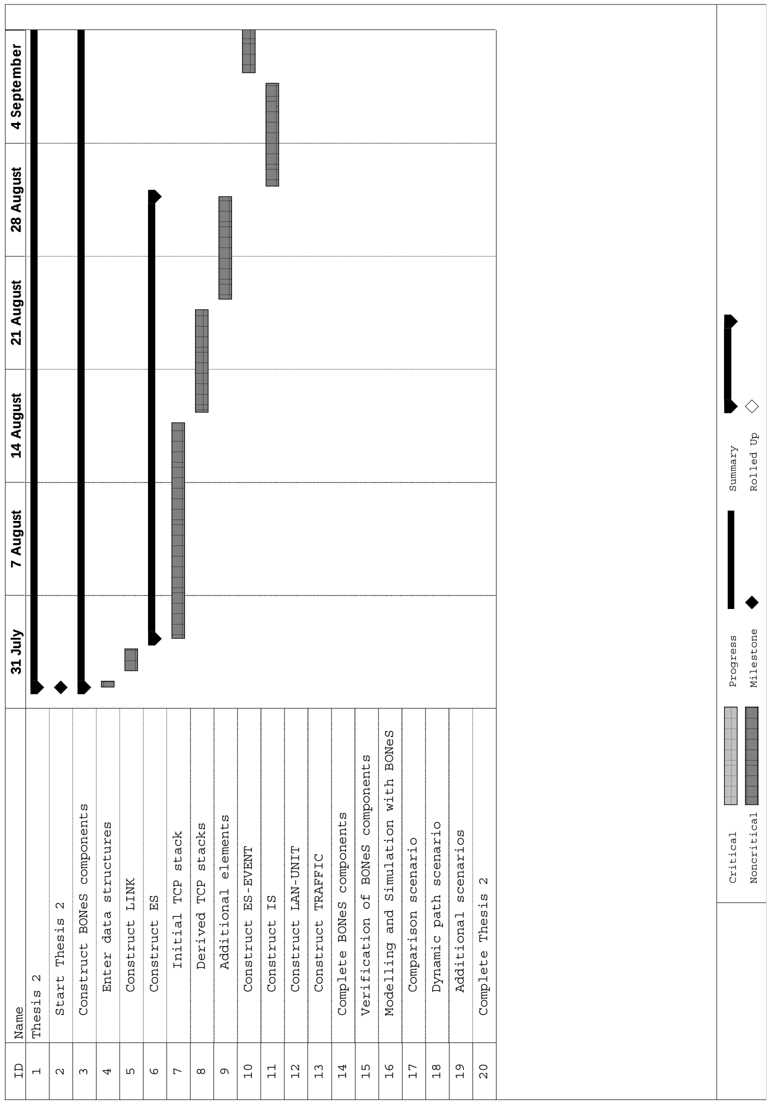

Figure 11: Gantt chart for Thesis 2 work – part 1

to see that the tasks are mostly sequential, however some of the construction tasks can be carried out in parallel.

5.3 Contingency plans

Consideration has been given the problems that may occur during the project. The following possible problems have been identified.

- It is possible that the estimation of time required to carry out tasks is too generous. My academic schedule is planned in such a way that I will be undertaking Thesis 2 during the Spring semester of 1995 in a full time capability with only 2 other 3 credit point subjects.

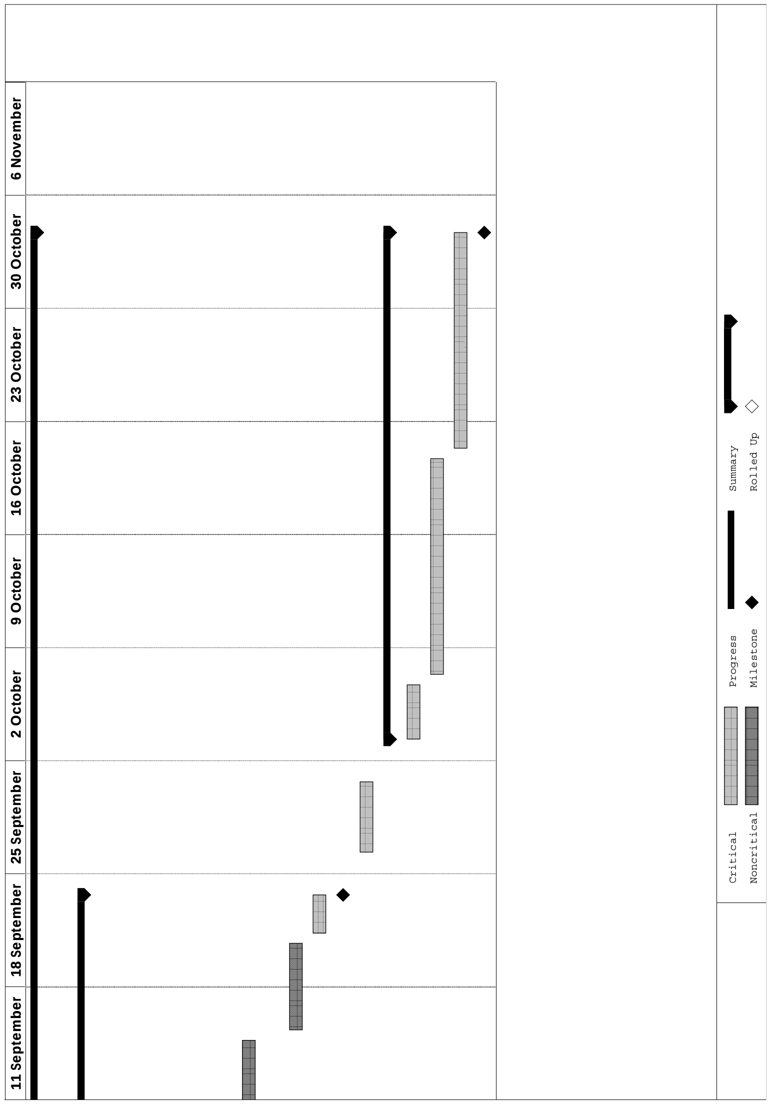

Figure 12: Gantt chart for Thesis 2 work – part 2

Firstly, I will attempt to complete at least one full component in BONeS before the commencement of the Spring 1995 semester. In doing this, I can get an idea of exactly how long it will take to complete the remaining components. The allocation of time to the project will be adjusted accordingly.

- Assumptions have been made about the capability of BONeS. Specifically it has been assumed that BONeS is capable of handling the large numbers of components planned for simulation runs. Examination of current BONeS models suggests that it can operate with models in the same order of magnitude being used in this project. The problem may lie in the amount of time it takes for BONeS to carry out the simulations.

  The only defence against this problem will be to run simulations overnight and on weekends when demands on computing resources are minimal.

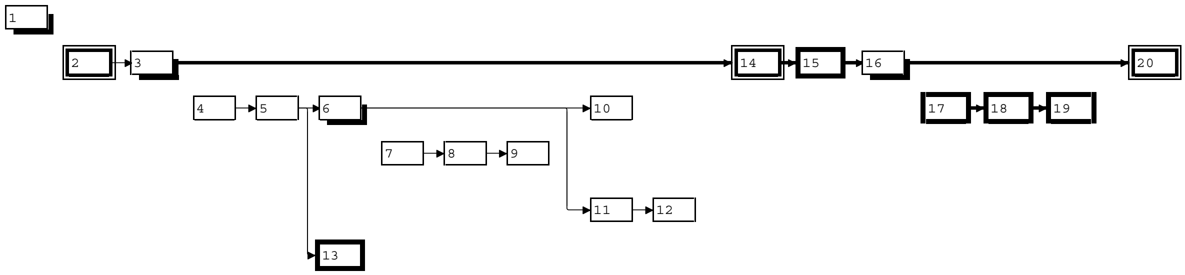

Figure 13: Network diagram (Task dependencies) for Thesis 2 work

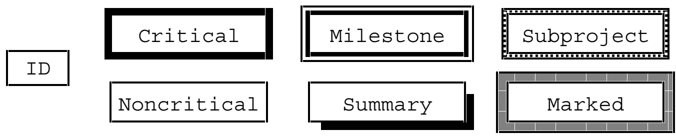

Figure 14: Network diagram legend

- To provide a safeguard against data loss and the inevitable 'human error' factor, the simulation library being used will be backed up according to the following schedule:

  Daily: Each day the simulation library is edited, any files changed will be backed up by copying them into a separate directory on the system. The previous 4 weeks worth or 4 copies of a file will be kept, whichever is the maximum – any extra will be deleted.

  This acts as a safeguard against accidental changes or deletions.

  Weekly: Each week the simulation library is edited, a copy of the entire library will backed up by copying it into a separate directory on the system. The backup will be retained online after which it is archived onto magnetic tape.

  This allows full copies of the library to revived in case of structural changes or loss of data on the system (unlikely).

- If other problems arise and it becomes clear that they are hampering progress of the project, I will attempt to move onto another task if it can be run in parallel. It was for this reason that the Network diagram previously shown is important.

6 Deliverables

The following items will result from the work carried out in Thesis 2,

- A BONeS library consisting of the components constructed above along with simulated models. These will be able to be called up, run and the results post-processed for display.

  The component library will be completed mid-way through the semester, however it is subject to revision if deficiencies are found when the models and simulations are run.

  The models and simulations will be completed by the end of the semester.

- The Thesis 2 report, describing the work performed and results obtained. This report will be constructed incrementally during the semester as the items of work are completed. The supervisor will be shown the progress of this report.

- Progress reports, by way of verbal meetings, will be given to the supervisor on a regular – weekly – basis. The purpose of these is to inform the supervisor that work is indeed being carried out, and any problems that may occur.

  I will keep a log book, detailing dates and work that has been carried out, this is both for my own use and to show that work has been done. The log book will be a basis for describing the current state of progress.

- The Thesis 2 presentation which is given at the end of the semester. The structure of this will mirror that of the report, however it will focus on the important issues and details from work carried out. I am planning to use colour overheads or video tape to show some of the work that was carried out using BONeS.

6.1 Thesis 2 report outline

The following is a possible rough outline of the report. It requires more examination before being finalised though.

1. Introduction
2. Congestion Avoidance and Control in wide-area TCP/IP
   (a) Description of the wide-area TCP/IP environment
   (b) Previous work on Congestion Avoidance and Control
   (c) Unresolved issues
3. BONeS as a simulation and modelling environment
   (a) Overview of the BONeS philosophy
   (b) Planning and constructing the library
   (c) Library components
4. Using BONeS to model and simulate Congestion scenarios
   (a) Comparison of existing schemes
   (b) Examining the effects of multiple (split) paths
   (c) *(others ...)*
5. Conclusion
6. References

References

Agrawala, A. K. and Sanghi, D. (1992). Design and evaluation of an adaptive flow control scheme, *Proceedings of the Conference on Computer Communications (IEEE Infocom)*, Vol. 3, IEEE, Florence, Italy, pp. 2391–2397 (11A.3).

Aslaksen, E. and Belcher, W. (1992). *Systems Engineering*, Prentice-Hall, Australia.

Braden, R. (1989). Requirements for Internet hosts – Communications layers, *Network Working Group Request for Comments RFC 1122*, Internet Engineering Task Force.

Brakmo, L. S., O'Malley, S. W. and Peterson, L. L. (??). TCP Vegas: New techniques for congestion detection and avoidance, *Technical report*, Department of Computer Science, University of Arizona, Tucson, AZ 85721.

Cáceres, R., Danzig, P. B., Jamin, S. and Mitzel, D. J. (1991). Characteristics of wide-area TCP/IP conversations, *SIGCOMM Symposium on Communications Architectures and Protocol s*, ACM, Zürich, Switzerland, pp. 101–112. also in *Computer Communication Review* 21 (4), Sep. 1991.

Chiu, D.-M. and Jain, R. (1989). Analysis of the increase and decrease algorithms for congestion avoidance in computer networks, *Computer Networks and ISDN Systems*, Vol. 17, pp. 1–14.

Clark, D. D. (1982a). Fault isolation and recovery, *Network Working Group Request for Comments RFC 816*, MIT Laboratory for Computer Science.

Clark, D. D. (1982b). Window and acknowledgement strategy in TCP, *Network Working Group Request for Comments RFC 813*, MIT Laboratory for Computer Science.

Comdisco Systems Inc. (1993). *BONeS DESIGNER Users Guide V2.1.*

Comer, D. E. (1991). *Internetworking with TCP/IP Vol II: Design, Implementation, and Internals*, Prentice Hall, Englewood Cliffs, NJ.

Comer, D. E. and Yavatkar, R. S. (1990). A rate-based congestion avoidance and control scheme for packet switched networks, *Proc. 10th International Conference on Distributed Computing Systems*, IEEE, pp. 390–397.

end2end-interest (1988+). End-to-end interest mailing list, end2end-interest@isi.edu.

Floyd, S. (1991a). Connections with multiple congested gateways in packet-switched networks. Part 1: One-way traffic, *Technical report*, Lawrence Berkeley Laboratory.

Floyd, S. (1991b). Connections with multiple congested gateways in packet-switched networks. Part 2: Two-way traffic, *Technical report*, Lawrence Berkeley Laboratory.

Floyd, S. (1994). TCP and Explicit Congestion Notification, *Technical report*, Lawrence Berkeley Laboratory.

Floyd, S. and Jacobson, V. (1992). On traffic phase effects in packet-switched gateways, *Internetworking: Research and Experience* **3**(3): 115–156.

Floyd, S. and Jacobson, V. (1993). Random early detection gateways for congestion avoidance, *IEEE/ACM Transactions on Networking* **1**(4): 397–413.

Hamburg, M. (1970). *Statistical Analysis for Decision Making*, Harcourt, Brace & World, Inc.

Huynh, L., Chang, R.-F. and Chou, W. (1994). Performance comparison between TCP slow-start and a new adaptive rate-based congestion avoidance scheme, *Proc. 2nd International Workshop on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems*, IEEE, pp. 300–307.

Jacobson, V. (1988). Congestion avoidance and control, *ACM Computer Communication Review* **18**(4): 314–329. Proceedings of the Sigcomm '88 Symposium in Stanford, CA, August, 1988.

Jacobson, V. and Braden, R. (1988). TCP extensions for long-delay paths, *Network Working Group Request for Comments RFC 1072*, Internet Engineering Task Force.

Jain, R. (1990). Congestion control in computer networks: Issues and trends, *IEEE Network Magazine*.

Jain, R. K. (1986). A timeout-based congestion control scheme for window flow-controlled networks, *IEEE Journal on Selected Areas in Communications* **SAC-4**(7): 1162–1167.

Jain, R. and Ramakrishnan, K. K. (1988). Congestion avoidance in computer networks with a connectionless network layer: Concepts, goals and methodology, *Proc. Computer Networking Symposium*, IEEE, pp. 134–143.

Kavi, P., Frost, V. S. and Shanmugan, K. S. (1991). Generic approach to LAN modeling, *in* B. L. Nelson, W. D. Kelton and G. M. Clark (eds), *Proc. 1991 Winter Simulation Conference*, IEEE, pp. 716–724.

Lamport, L. (1986). LaTeX: *A Document Preparation System*, Addison-Wesley.

Liebeherr, J. and Akyildiz, I. F. (1990). Performance analysis of gateways with buffer constraints, *Proc. IEEE INFOCOM 90*, Vol. 3, IEEE, pp. 1204–1212.

Maltby, G. (1992). An introduction to TeX and friends.

Mankin, A. and Ramakrishnan, K. (1991). Gateway congestion control survey, *Network Working Group Request for Comments RFC 1254*, Internet Engineering Task Force.

Miller, I., Freund, J. E. and Johnson, R. A. (1990). *Probability and Statistics for Engineers*, 4 edn, Prentice-Hall.

Mogul, J. C. (1992). Observing TCP dynamics in real networks, *SIGCOMM Symposium on Communications Architectures and Protocol s*, ACM, Baltimore, Maryland, pp. 305–317. *Computer Communication Review*, Volume 22, Number 4.

Munro-Smith, B. and Farhangian, K. (1993). The performance evaulation of wireless LAN systems, *IEE Colloquium on 'The cordless Office'*, IEE, pp. 1/1–12.

Nagle, J. (1984). Congestion control in IP/TCP internetworks, *Request for Comments RFC 896*, Internet Engineering Task Force.

Nagle, J. (1987). On packet switches with infinite storage, *IEEE Transactions on Communications* **COM-35**(4): 435–438.

Nicholas, J. M. (1990). *Managing business and engineering projects: concepts and implementation*, Prentice-Hall.

Paxson, V. (1991). Measurements and models of wide area TCP conversations, *Technical report*, Department of Computer Systems Engineering, University of California, Berkeley, CA.

Paxson, V. (1993). Growth trends in wide-area TCP connections, *Technical report*, Lawrence Berkeley Laboratory.

Postel, J. (1981a). Internet Control Message Protocol, *Network Working Group Request for Comments RFC 792*, Information Sciences Institute, University of Southern California.

Postel, J. (1981b). Internet Protocol, *Network Working Group Request for Comments RFC 791*, Information Sciences Institute, University of Southern California.

Postel, J. (1981c). Transmission Control Protocol, *Network Working Group Request for Comments RFC 793*, Information Sciences Institute, University of Southern California.

Postel, J. (1983). The TCP maximum segment size and related topics, *Network Working Group Request for Comments RFC 879*, Information Sciences Institute, University of Southern California.

Robinson, J., Friedman, D. and Steenstrup, M. (1990). Congestion control in BBN packet-switched networks, *Computer Communication Review* **20**(1): 76–90.

School of EE UTS (1994). *UTS Electrical Engineering Thesis Subjects*, A/94 edn.

Shanmugan, K. S., LaRue, W. W., Klomp, E., McKinley, M., Minden, G. J. and Frost, V. S. (1988). Block-Oriented Network Simulator (BONeS), *Conference Record, GLOBECOM 88*, Vol. 3, IEEE, pp. 1679–1684.

Sidhu, D. P. and Blumer, T. P. (1985). Some problems with the specification of the military standard Transmission Control Protocol, *Request for Comments RFC 964*, SRI International, Menlo Park, California.

Spragins, J. D., Hammond, J. L. and Pawlikowski, K. (1992). *Telecommunications : Protocols and Design*, Addison-Wesley.

Stallings, W. (1991). *Data and Computer Communications*, 3 edn, Macmillan Publishing.

Stallings, W. (1993). *Networking Standards : A guide to OSI, ISDN, LAN and MAN standards*, Addison-Wesley.

Thompson, D. R., Moorhead, R. J. and Smith, W. D. (1992). Advanced Solid Rocket Motor (ASRM) communications network analysis using BONeS, *Proc. IEEE SOUTHEASTCON 92*, Vol. 1, IEEE, pp. 276–282.

Tipper, D., Hammond, J. L., Sharma, S., Khetan, A., Blakrishnan, K. and Menon, S. (1994). An analysis of the congestion effects of link failures in wide area networks, *IEEE Journal on Selected Areas in Communications* **12**(1): 179–192.

Ulusoy, Özgür. and Baray, M. (1988). Dynamic congestion control in interconnected computer networks, *Proc. 13th Conference on Local Computer Networks*, IEEE.

Wang, Z. (1992). *Routing and Congestion Control in Datagram Networks*, PhD thesis, University College London.

Wang, Z. and Crowcroft, J. (1991). A new congestion control scheme: Slow Start and Search (Tri-S), *Computer Communication Review* **21**(1): 32–43.

Wang, Z. and Crowcroft, J. (1992). Eliminating periodic packet losses in the 4.3-Tahoe BSD TCP congestion control algorithm, *ACM Computer Communication Review* **22**(2): 9–16.

Zhang, L., Shenker, S. and Clark, D. D. (1991). Observations on the dynamics of a congestion control algorithm: the effects of two-way traffic, *Sigcomm '91 Conference: Communications Architectures and Protocols*, ACM, Zürich, Switzerland, pp. 133–147.

Zheng Wang, J. C. (??). A dual-window model for flow and congestion control, *Technical report*, Dept Computer Science, University College London.