DEPARTMENT OF DEFENCE
DEFENCE SCIENCE & TECHNOLOGY ORGANISATION
DSTO

DSTO

A Transmission Availability Forecast Service for Internet Protocol Networks.

W.D.Blair and R.Jana

DSTO-RR-0146

A Transmission Availability Forecast Service for Internet Protocol Networks.

W.D. Blair and R. Jana

Communications Division
Electronics and Surveillance Research Laboratory

DSTO-RR-0146

ABSTRACT

The quality of available network connections, especially the bandwidth available to clients using the connections, has a large impact on the performance of distributed applications. For example, document transfer applications such as the World Wide Web suffer a dramatic increase in response times as a result of network congestion causing a reduction in the available bandwidth of the connection. This paper recognises the explosion of interest in the use of Internet Protocol (IP) networks within the Australian Defence Organisation and describes the development of a software tool for estimating the available bandwidth between a server and client in a distributed computing environment. We discuss the design and implementation details of the Transmission Availability Forecaster (TAF) probe and present validation studies demonstrating its reliability and accuracy in the context of actual Internet conditions.

Published by

DSTO Electronics and Surveillance Research Laboratory
PO Box 1500
Salisbury South Australia 5108 Australia

Telephone: (08) 8259 5555
Fax: (08) 8259 6567

AR-010-824
February 1999

APPROVED FOR PUBLIC RELEASE

A Transmission Availability Forecast Service for Internet Protocol Networks.

Executive Summary

The increasing popularity of distributed information services like the World Wide Web has resulted in a number of intriguing issues. One important characteristic of a network connection is the bandwidth available to clients using that connection. For document transfer applications higher bandwidth availability implies faster document transfer time. Available bandwidth depends on 1) the capacity of the path between client and server, limited by the slowest or (bottleneck) link speed, and 2) the presence of background or competing traffic, i.e. congestion.

This report describes the Transmission Availability Forecaster (TAF) service implemented on DSTO's Experimental Command Control Communications and Intelligence System Technology Environment (ExC3ITE) testbed for a best effort IPv4 network. TAF makes no assumptions as to support available from the network management system (for instance in providing traffic statistics) and seeks to remain autonomous and not reliant on network management support. Thus the tool will have application beyond the private ExC3ITE network.

The fundamental TAF design criteria were:
- The service should be as quick as possible;
- The technique should not add undue additional traffic to the network; and
- The estimate should tend to be conservative (ie pessimistic).

The challenge of predicting performance in a dynamic network was acknowledged by the ExC3ITE developers. Nevertheless, the provision of a performance estimate of the same order of magnitude as would occur, or a good estimate of the range of likely performance, would still be a valuable service in the absence of any other guidance.

To address these difficulties, the current TAF provides: a minimum predicted bit rate (pessimistic estimate); a maximum predicted bit rate (a relatively optimistic estimate); and a likely expected bit rate.

This report describes a basic Transmission Availability Forecaster service. Trials to date have shown it can successfully reduce the range of uncertainty of available capacity from perhaps 4000:1 to under 17:1 (and typically better than 4:1). The probe is intended to be used immediately before a download and so will take account of long term changes in the network situation. The probe measurement takes a finite period and so can aggregate and characterise short term variations in network situation. Nevertheless, the process remains vulnerable to medium term variations, ie changes that occur after the probe and before the download.

While the TAF service has been developed as a CORBA tool on ExC3ITE, it could be fielded on any IP network. It would provide considerable utility on Defence networks where there is a large range of capacity in the network, and where predictable performance is desirable.

Authors

W.D. Blair
Communications Division

William Blair is an Electronics Engineer in Network Integration Group of the Defence Science and Technology Organisation's (DSTO) Communications Division having joined DSTO from the Australian Army. He is involved in research into networking techniques to control Quality of Service (especially Asynchronous Transfer Mode - ATM) for military strategic and tactical communications.

R. Jana
Communications Division

Rittwik Jana is a Professional Officer with the Network Integration Group of the Defence Science and Technology Organisation's (DSTO) Communication Division. His main research interests include computer networking and robust receiver architectures for spread spectrum systems. He is currently pursuing a PhD in telecommunications at the Australian National University.

Contents

1. Introduction

There has been an explosion of interest and use of Transmission Control Protocol/Internet Protocol (TCP/IP) networks within Defence. While the adoption of IP makes it easier to use commercial software on the network, it has some characteristics that impact on its suitability for military use. Two related examples are the lack of guarantees of network performance to users as well as the absence of indications of performance that can be anticipated. These are issues being explored in the Defence Science and Technology Organisation (DSTO) within its Experimental Command Control Communication and Intelligence (C3I) Technology Environment known as ExC3ITE.

The absence of performance measurement and control mechanisms is an issue of particular concern to one specific project on the ExC3ITE - the Imagery Management and Dissemination (IMAD) project. IMAD will be disseminating large images thus its performance will be significantly impacted upon by transmission availability. IMAD anticipates two services on ExC3ITE: Transmission Availability Forecasting (TAF) and Capacity Reservation.

This report describes the TAF Service implemented on ExC3ITE for the best effort IPv4 network. TAF makes no assumptions as to support available from the network management system (for instance in providing traffic statistics) and seeks to remain autonomous and not reliant on network management support. Thus the tool will have application outside the private ExC3ITE network.

In Section 2 we provide some background to this work. Section 3 describes the architecture of the TAF system in particular the concept of TAF Managers controlling TAF Agents distributed across the network. In Section 4 we describe the TCP/IP protocol and how it is modelled in the TAF system as two components. Section 5 describes the implementation and performance of the TAF bandwidth estimator while Section 6 examines how the TAF implementation is offered as a CORBA service on ExC3ITE. Section 7 concludes and considers possible further work in this area.

2. Background

2.1 ExC3ITE Architecture

DSTO is currently developing the ExC3ITE to facilitate technology development and demonstration. The network comprises three core locations and five re-locatable sites. ATM technologies provide the basis of the network, with an IP network operating across the ATM. There are a series of applications running over the network and these are seeking to define the next generation Defence C3I architecture with a specific interest in distributed systems and the management of network resources. The IMAD project in particular is promising to influence the early development of ExC3ITE. Note

that while the TAF and capacity reservation service would be available for applications operating in a native ATM mode on ExC3ITE, these need to be built for the IP based applications.

2.2 IMAD

The IMAD system is a technology demonstrator that will implement and demonstrate a federated imagery management system using the Common Object Request Broker Architecture (CORBA). CORBA provides an environment for developing and deploying object based distributed applications. The key components of IMAD, illustrated in Figure 1, are:

- image libraries consisting of digital geospatial images produced from aerial photography or earth observation satellites;
- the Image Query Manager (IQM) which is responsible for handling queries spanning multiple distributed libraries;
- a Context Manager, in charge of determining the best possible method to transfer the image from a particular library to the end user;
- other specialised services such as TAF Managers, Traders, Compression and Tiling services; and finally
- the CORBA backbone used to glue these distributed servers and clients together.

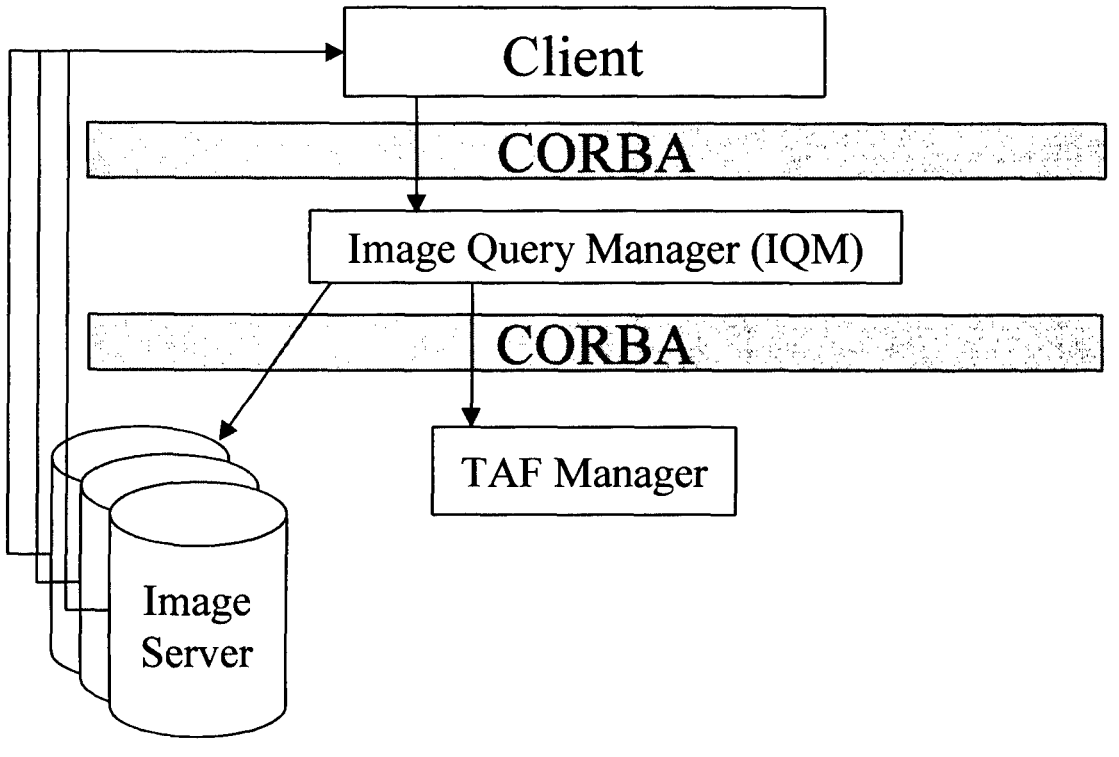


Figure 1 - Key Components relevant to IMAD and TAF.

It is important to note that in IMAD, interaction between these distributed servers and clients occurs via the Object Request Broker (ORB). The communication within the ORB can be visualised as happening at a higher protocol level than the internet layer or the transport layer. Although the transparency of these services is one of the more powerful aspects of CORBA, it incurs a significant performance overhead. Accordingly, the actual transfer of the image will occur using the IP file transfer protocol (FTP) or via a raw TCP stream type connection over IP.

There are a number of research issues associated with the development of ExC3ITE and the IMAD system will provide a platform for this research. Further details on IMAD are contained in Grigg et al [1].

3. TAF Concepts

3.1 Fundamentals

In this report we formulate a model that estimates the available bandwidth and hence the network component of the image transfer time on a best effort IPv4 network. The overall aim is to deliver predictable performance to its users. The use of CORBA in the IMAD architecture leads to a requirement for a CORBA interface to the TAF service. Although it is not essential for the TAF to rely on a CORBA based architecture, this is seen as a "value add" for potential future distributed applications in that allows use of the TAF service in a seamless and an integrated fashion.

The fundamental TAF design criteria were:

- The service should be as quick as possible;
- The technique should not add undue additional traffic to the network; and
- The estimate should tend to be conservative (ie pessimistic).

The challenge of predicting performance in a dynamic network was acknowledged by the ExC3ITE developers. Nevertheless, the provision of a performance estimate of the same order of magnitude as would occur, or a good estimate of the range of likely performance, would still be a valuable service in the absence of any other guidance.

The measurement involves initiating a probe from source to destination and measuring a number of relevant statistics characterising a model of the IP connection. These are then used in a TCP model to determine an estimate of network utilisation. The client is finally advised of the quality of the connection to the candidate server(s). The application may choose to retrieve the document from the server with the highest quality connection. Our prime motivation is the dissemination of large images in a distributed network, thus performance will be significantly impacted upon by transmission availability.

It is now appropriate to discuss briefly some of the work done by various researchers which has contributed to the development of TAF. The problem of estimating the quality of a link dates back to 1980 when the well known "ping" probe was devised [2]. This measured the source-destination Round Trip Time (RTT). To date, this diagnostic tool has been the most common and the quickest method to detect if a link is available. Soon afterwards, Jacobson [3] realised that a tool to enumerate the links in a source-destination path would be very useful to find out the number of hops a train of IP packets would traverse. It relied on the Internet Control and Management Protocol (ICMP) "time exceeded" time stamp to be relayed across the network back to the original sender for each hop. In 1997 a software tool named "pathchar" [4] was constructed to infer the characteristics of Internet paths. This tool attempted to unify the functionalities of both "ping"and "traceroute". By continuously bombarding the IP network with a train of ICMP packets, it can determine the link speed for each hop. Although this tool has tried to capture the essence of bandwidth availability, it takes a long time for a complete analysis of a network with many hops. In time critical missions, one cannot afford to have the luxury of waiting too long before a decision is made. This motivated us to design and implement a lightweight and robust probe.

As will be discussed later, the time taken for a packet to traverse the path is a significant factor in determining the bit rate for a connection. One can consider this time to comprise two fundamental elements:

- an element related to the processing and queuing delays occurring in the store and forward action of intermediate routers; and
- transit delays determined by the speed at which the routers can clock the packets onto outgoing links at the bit rate of the underlying physical connection.

While an IP network can in principle provide different routes through the network for each packet sent on a user to user link, normally routes are relatively stable. Accordingly, the transit delay element will remain relatively static. By contrast, the queuing delays are highly dynamic with variations in a range of time scales upwards from sub-second to seasonal or yearly. One can consider that, leaving aside the characteristics of the data communications protocols, the capacity available to a connection will be a portion of the underlying physical bit rate limit, where this portion will be determined by the amount of competing traffic evidenced by the queuing delays. In relation to this idea, "ping" can only provide the aggregate delay. The "pathchar" program is specifically measuring each transit delay element and sees variations in the processing and queuing delays as noise in this effort. TAF seeks to characterise both elements of the delay.

Consideration of the dynamics of internet traffic, including its self-similarity, is an on-going and challenging research topic which we sought to avoid. Instead the TAF concept was built around a tool that undertakes a relatively short examination of the network immediately before the client commences downloads. The finite period of examination will tend to aggregate and characterise the short term dynamics of the situation while the on-demand probing will take account of the longer term variations

in network situation. TAF remains vulnerable to the medium term changes that might occur between the completion of the examination and the commencement of the download.

3.2 TAF Architecture

The ExC3ITE network is a mix of IP and ATM networks interconnected over media such as twisted pair, optical fibre and satellite. Bandwidth and delay characteristics of a particular connection will depend on the path taken through the network as well as the characteristics of TCP/IP.

A typical network scenario can be seen in Figure 2. Images can be situated in different libraries/servers. Clients anywhere in the WAN use a Web browser to search for these images by connecting to an IQM; later access to these images is direct. The IQM obtains bandwidth availability forecasting, from a single point of contact - the TAF Manager - for advice to the client. This manager can be located anywhere in the network and is responsible for estimating the conditions that would be experienced by a connection between the client and a specified image library.

In most networks, including those based on IP, it is difficult for a third party, ie a party that is neither source nor destination of a connection, to determine the performance of that connection. Accordingly, the TAF Service requires that a TAF Agent share the same LAN as each image library. A performance prediction between such an agent and the client should be well correlated with the performance between the associated image server and the client. Each agent is controlled by and communicates information with a TAF Manager.

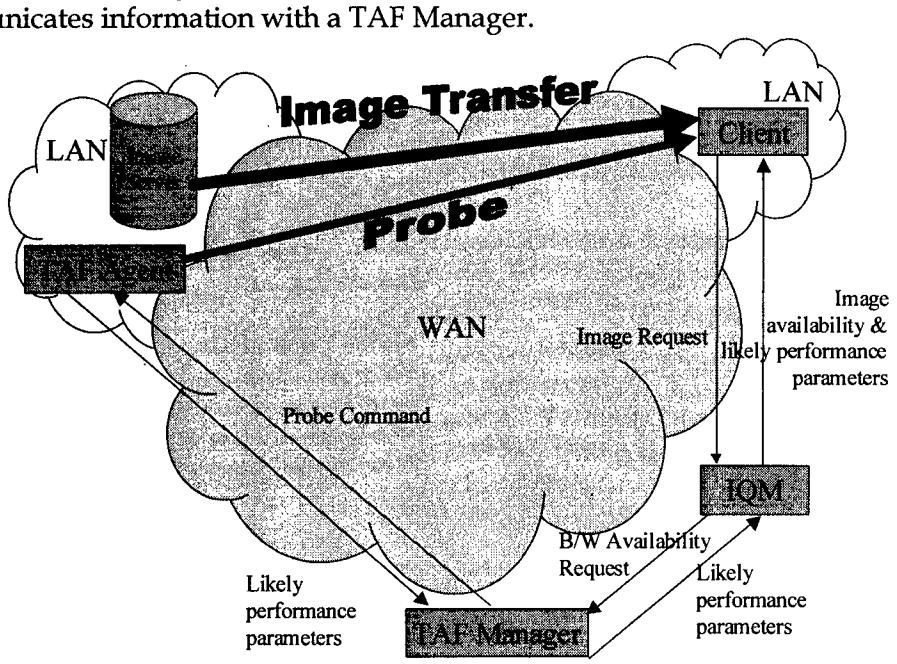


Figure 2 – Concept of a third party initiated probe.

The TAF Agents measure appropriate network parameters between the Image Server LAN and the client. These parameters are used to calculate the likely range of available bandwidth on the link and a best estimation of likely bandwidth. These are passed back to the IQM through the TAF Manager so that they can be associated with each image queried.

4. The TAF Model of TCP/IP

In this section we describe some the fundamental concepts behind TCP/IP communications leading to the TAF model of the protocol.

4.1 TCP/IP Characteristics

For the purposes of the analysis being carried out by the TAF, TCP/IP is a family of protocols operating over two fundamental levels of the layered model of data communications. Further details can be found in Tanenbaum [5].

- Internet Layer. The protocol operating at this layer is IP. This provides a best effort datagram service from one computer host to another. The datagrams (or packets) comprise portions of the user information stream that has been broken up for forwarding through the network. No guarantees are made that packets indeed arrive at the destination (hence the term "best effort") nor to control congestion by controlling the amount of data placed on the network.
- Transport Layer. There are two protocols at this layer, TCP and User Datagram Protocol (UDP), but for typical communications TCP is the one used. TCP is responsible for segmenting user data streams and passing these segments to the IP layer for forwarding. In addition, TCP entities at source and destination communicate via an acknowledgment scheme to ensure that the segments are delivered in sequence and without error. Finally, TCP is responsible for controlling the flow of segments onto the network to avoid/respond to network congestion. There is no explicit advice on network status from the network to the TCP entities, thus the TCP entities have to deduce this status via observation of data packet transmissions. The algorithms for deducing network status, and determining TCP's reaction, have a significant impact on the effective throughput that will be achieved.

The TAF model is implemented as the interaction between two models: one operates at the IP layer and the other at the TCP layer. The IP model considers the passage of IP packets through the network. Parameters derived from this model provide the input to the TCP performance model.

4.2 IP Model

The TAF IP model, shown in Figure 3, sees the link as comprising two elements that delay the passage of packets from source to destination as measured by the one-way transit time (OTT):

- The first delay element, T_{PQ}, is conceptually independent of the speed of the network links and length of the packet. It encompasses all the delays related to the processing of each packet as well as delays caused by the presence of buffered packets ahead in the queue awaiting access to the transmission link.
- The second delay element, T_T, is conceptually the transmission link and is reflected in the time taken to clock the bits that comprise the packet onto the link. Note that in a multi-hop/store and forward path T_T would strictly be the sum of the reciprocals of all bit rates in the path multiplied by the packet length. If one link were substantially slower than the others though, it would dominate the calculation. In this case the T_T for the entire path would substantially be the T_T through the slowest link (bottleneck) of the path from source to destination.

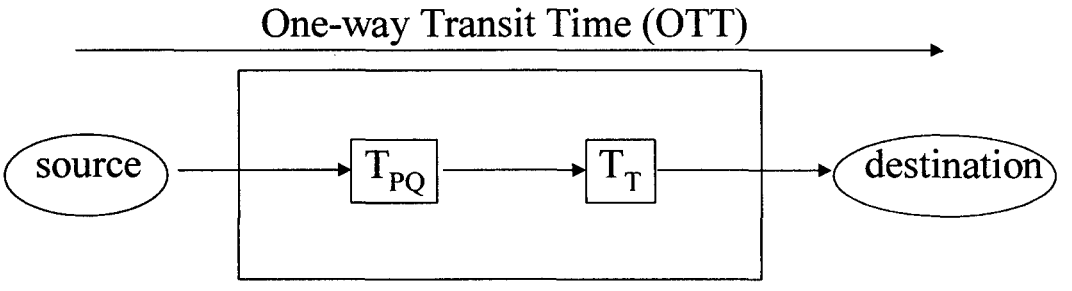


Figure 3. TAF IP Model

Since T_{PQ} encompasses all overhead delays associated with the packet, it also includes the transmission time for any packet headers. All else being equal, if the length of the packet payload is increased then the OTT will increase, because of the increase in T_T, by an amount equal to the time taken to clock through the number of extra bits. Figure 4 shows how the raw bottleneck bit rate can conceptually be measured. Note there is an underlying assumption that the raw bottleneck bit rate is constant, set by the rate of the physical media over which the traffic is being passed. Variable bit rate media, as can be provided by ExC3ITE's ATM underlay, will be addressed in future TAF developments.

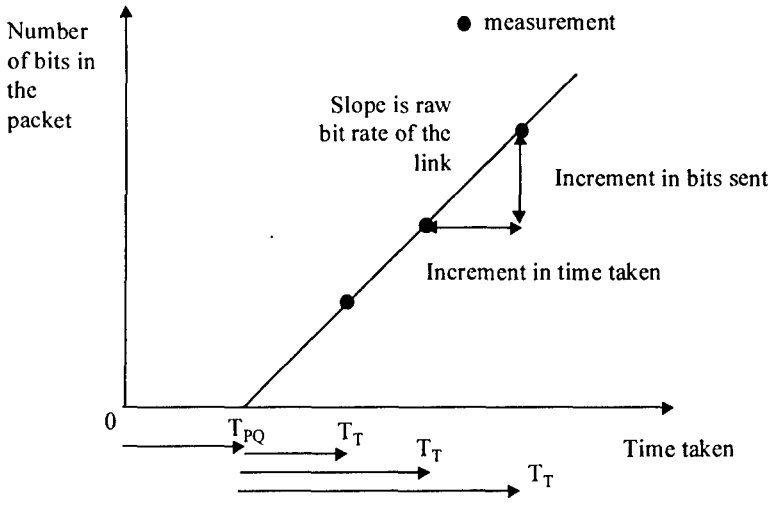


Figure 4. OTT versus Number of Bits Sent

4.3 TCP Model

The principal role of TCP is to provide a reliable transfer (ie no errors or changes in sequence) of packets of data from source to destination. As part of this function, TCP attempts to be a well-behaved user of the network by seeking to avoid creating network congestion and reacting in a responsible fashion to perceived network congestion - this is the role of the flow control aspect of TCP.

The error checking function of TCP is fulfilled by a 16 bit checksum of the TCP packet. In the event of an error, the packet is discarded and treated as if it never arrived. Clearly, errors will have a significant impact on the performance of TCP, however, for the purposes of this version of the TAF, it is assumed that the network is providing error free transmission. This is a reasonable assumption for the nature of the ExC3ITE network, and can be relaxed in future versions of the TAF.

Guaranteed complete and in-sequence flow of packets is provided via an acknowledgment process. TCP employs a sliding window, ie at any moment there can be a set maximum amount of data that has been transmitted without the sender having yet received an acknowledgment. As early packets of data are acknowledged, the sender is free to transmit additional packets, effectively advancing the window through the stream of data to be sent. The acknowledgment and sliding window elements are inexorably bound to the congestion mechanisms. Some points to note are:

- When each packet is transmitted, a timer is started. When the timer expires the packet is assumed to have been lost because of congestion. In the generic versions of TCP, the packet and **all subsequent packets that have previously been sent** will then be queued for retransmission. Clearly this can significantly reduce the effective throughput of the system.

- The setting of the retransmission timer is an adaptive process that in effect attempts to estimate the longest time a packet acknowledgment could reasonably be expected to take. This is a delicate balance of not wanting it to be too short (leading to unnecessary timeout and resultant redundant and wasteful retransmissions) or too long (unnecessary delay before initiating valid retransmissions). The process typically uses an exponential smoothing algorithm that slowly responds to trends in the RTT.
- The size of the sliding window, called the (transmit) congestion window, varies depending on the state of the network connection. In the event of acknowledgments being received inside the timeout, the congestion window will grow (in a manner misleadingly known as "slow start") providing greater utilisation of the network. The window cannot grow larger than the lesser of two parameters: the maximum window size parameter in the sending machine and the available buffer size periodically advertised by the receiving machine. If a timeout occurs, the congestion window is immediately reduced to one packet (in the generic version of TCP) and the slow start recommenced.
- In a well-behaved network at steady state, the sending machine congestion window will have reached its maximum value. New packets will be sent on receipt of each acknowledgment. The effective throughput will be given by:

$$throughput = \frac{congestion\ window}{RTT}$$

Another way of envisaging this calculation is as follows. If there are n packets in a congestion window and packets are sent at equal intervals of RTT/n; then by the time the first packet is acknowledged (ie RTT after it was transmitted) there will be exactly one congestion window sent but not yet unacknowledged. This would mean effective throughput is:

$$throughput = \frac{packet\ size}{packet\ period}$$

or

$$throughput = \frac{\frac{congestion\ window}{n}}{\frac{RTT}{n}}$$

or as before:

$$throughput = \frac{congestion\ window}{RTT}$$

Further details can be found at Tanenbaum [5].

5. The TAF Implementation

5.1 IP Probe

In this section we describe some of our test experiments and the approaches taken to estimate the parameters modelling the IP layer.

5.1.1 Software Approach Evolution

Since the client software will not have any TAF functionality (other than knowledge of the TAF CORBA interface), the TAF Agent must operate with limited co-operation from the client host. Three options for client co-operation were considered based on standard facilities that can be expected on any host running TCP/IP:

- response to ICMP echo requests (the Ping response daemon),
- the operation of the UDP echo port, and
- the operation of the TCP echo port.

The use of the TCP echo facility is not appropriate to IP modelling and examination was reserved until the TCP modelling effort. There was some concern that ICMP packets were treated by the network differently to normal IP packets, as ICMP requires additional processing. A brief analysis on DSTO's corporate network showed that there was insignificant difference in time delays experienced by ICMP versus IP packets using the UDP echo service. Since UDP echo services are sometimes disabled for security reasons, the work continued on the basis of using ICMP echo requests.

Appendix One provides more detail on the IP probe, however the approach can be summarised as follows:

- ICMP echo requests are sent in a series of short bursts of multiple packets,
- the data from the first burst is discarded as timings are often distorted by additional processing delays such as routing table lookup,
- the gap between return times of the packets in the subsequent bursts provide the measurement of the T_T,
- the RTT of the first packets of each burst, assumed to be twice the OTT, is used to determine T_{PQ} given knowledge of T_T, and
- the packet size in the bursts is varied and the change in measured T_T caused by the change in payload bits provides the measurement of the bottleneck bit rate.

5.1.2 Probe Performance

An early finding was that the DSTO network was operating with payload and header compression activated. This resulted in artificially high measured data rates. Since IMAD will be disseminating compressed images, it would not benefit from any router based compression. Accordingly, the IP probe carries a random data payload, which

does not exhibit this increase in effective bit rate. The compression of IP and physical headers is encapsulated in T_{PQ}.

Trials of the IP probing process were initially conducted on DSTO's corporate network between Fern Hill Park (Canberra) and Salisbury (Adelaide). This comprised the minimal link scenario of two LANs joined together by a single router to router connection, in this case a 128kbps ISDN line. In this simple scenario, the IP probe was very effective in determining the bit rate of the bottleneck and measures of the queuing delay in the routers. The results were sufficiently encouraging for us to move on to the TCP modelling effort and TCP prediction trials.

During the TCP modelling and bit rate prediction trials, probes (and subsequent validation downloads) were conducted over three link scenarios: local (three hops), national (nine hops) and international (23 hops); and three different file sizes: small (10 kbytes), medium (1 Mbytes) and large (3 Mbytes). The trial results can be summarised as:

- Local. Performance was as expected from previous trials.
- National. This was the most difficult environment. There were large variations in T_{PQ} and estimated bottleneck link rate. As this test crossed from the Defence intranet to the worldwide Internet, the Defence firewall may have contributed with perturbations in T_T measurements.
- International. This scenario was more encouraging as the long T_{PQ} with only small variation led to a more stable estimate of One-way Transit Time.

5.2 TCP Prediction

In this section we describe some of our test experiments and the approaches taken to model the TCP layer.

5.1.2 Software Approach Evolution

The IP model gives T_{PQ} and T_T (or more strictly the T_{PQ} statistics and the link bottleneck bit rate). This can then be used to predict the OTT of a packet of any size. Packet sizes are determined by the physical media, but 1400 bytes is typical (this is the data payload size for ethernet and point-to-point protocol links). Unlike the probe packet which is echoed back intact, TCP acknowledgments are either zero payload packets or piggyback on other data; accordingly the TCP model estimates RTT to be $T_T + 2xT_{PQ}$ (T_T for data payload size rather than a probe packet).

In principle one can then estimate the effective capacity of the connection using this RTT and the equation in paragraph 4.2, however two difficulties are encountered:

- The RTT estimate does not take account of self-congestion. The T_{PQ} figures are good estimates for isolated packets. However, when a window is transmitted in a burst, early packets will queue and increase the T_{PQ} experienced by later packets.

- The maximum receive buffer size is a parameter that can be adjusted on each host. Indeed this can be adjusted under program control - typically, TCP sockets operate with an 8kbyte maximum buffer, whereas FTP often operates with a 24kbyte buffer. This cannot be determined via the probe.

To address these difficulties, the current estimator makes the following assumptions in deriving performance predictions:

- The minimum predicted bit rate provides a conservative estimate of possible performance. It assumes that the isolated packet RTT can be well estimated from the IP model and TCP moves one 1400 byte packet each RTT. Note that the RTT is calculated using the maximum T_{PQ} to accord with our pessimistic estimator criterion.
- The maximum predicted bit rate provides a relatively optimistic estimate of possible performance. It assumes that there is a constant 24kbyte receive window and no change from the isolated packet RTT estimate. Note that if this figure exceeds the measured bottleneck bit rate, the model will respond with the bottleneck bit rate as the estimated maximum instead of this "receive window limited" bit rate.

At this point it should be noted that the TAF implements two checks prior to undertaking the full probe. If the early probes are indicative of a very low bottleneck bit rate (currently set for measurements less than 2.4 kbps such as might be seen from a narrow band satellite link) then continuation of the probe would lead to undue congestion on this scarce resource. Accordingly, the TAF will cease probing and return nominal low figures for the bit rate predictions. By contrast, extremely high bit rates (currently defined as greater than 8 Mbps) are indicative of there being no clear bottleneck, for instance when client and server are on the same LAN. At this time, TAF will cease probing and respond with a nominal high bit rate as being available.

The ratio of the maximum to the minimum predicted bit rate from the TCP model ranges up to 17:1. While this is a considerable improvement on the 4000:1 (8 Mbps: 2.4 kbps) uncertainty without TAF, we considered that we needed to develop a tighter estimate of the achievable bandwidth. Thus, the third performance parameter calculated by the TCP model is a best estimate of the likely bit rate within the range described by the minimum-maximum predicted bandwidth.

At Appendix One there is a description of the examinations made in developing this estimator which was titled the Normalised Estimated Bandwidth. The Normalised Estimated Bandwidth is a linear measure giving a figure between 0 and 1 indicating where the achieved bit rate occurred in the range of minimum expected bit rate (where the NEB would be 0) to maximum expected bit rate (where the NEB would be 1).

The product of the mean T_{PQ} and the bottleneck bit rate is indicative of the mean number of bits in the bottleneck queue. The NEB was found to be quite strongly (inversely) correlated with this figure. With larger "queue" figures actual performance

tended towards the minimum predicted figure, while small figures were indicative of actual performance close to the maximum predicted. The product was then considered as the basis of estimating the likely bit rate in the production TAF service. Trials were conducted to determine a suitable formula for predicting NEB.

5.2.2 TCP Estimator Performance

The Estimated Bit Rate predictor performs quite well. In 27 trials of the total TAF, eight failed to produce a conservative predicted likely bit rate, ie the actual performance was slower than predicted. Four of these failures were from the problematic 10 kbyte downloads. We then considered a range around the predicted likely bit rate defined by a high limit of two times the predicted likely bit rate and a low limit of one half the prediction. On two occasions (again the problematic 10 kbyte downloads) the experienced bit rate was less than half the predicted rate. Of less concern given our goal of pessimistic estimates, on two occasions the experience bit rate exceeded twice the predicted rate. The range bounded all the other trials. This performance will be sufficiently accurate for the intended application and should still provide sufficient utility for other applications. The results of the trials are shown in the scatter diagram as Figure 1-8. Scatter diagrams are interpreted by considering each point as a trial. The x-axis value is the TAF predicted bit rate and the y-axis value is the achieved bit rate.

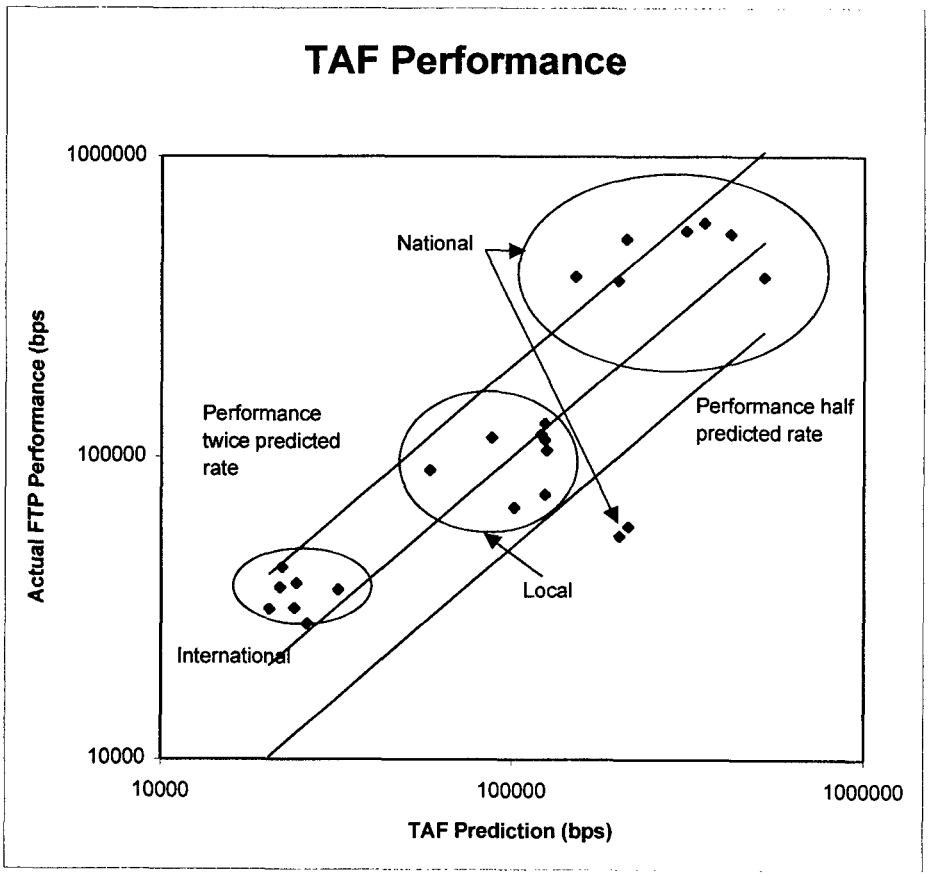


Figure 5 TAF Prediction Performance

The reasons for the failures in the prediction are seen to be straightforward:

- The 10 kbyte file downloads are difficult to predict because, being short, they will suffer from short term disruptions to the network load.
- The local trials had some prediction failures as the number of users of the link was limited. As a consequence the addition or subtraction of a single shared user from the link has a significant impact on the capacity available to the remaining users. Regardless, the performance of the TAF retains considerably utility.

5.2.3 Outstanding Issues

Beyond the model itself and into the client software, the potential for a client to download multiple images simultaneously will need to be handled in some fashion. At this stage, the TAF Service does not take account of the potential for multiple downloads. There needs to be some mechanism to determine if the bottleneck between clients and different servers is actually the same bottleneck (simultaneous downloads will share the capacity) or different (potential for each download to use the full bandwidth). Until this is done, client software should anticipate that the capacity advised by TAF Manager is equally shared between the downloads

6. CORBA Aspects of the Implementation

The actual implementation of the TAF Service on ExC3ITE is slightly more complicated than described in Figure 2. There are multiple TAF Managers - running on each major image server LAN whilst the TAF Agents are not separate objects, but are embedded into each TAF Manager.

Each TAF Manager registers itself to the CORBA trader. Each registration includes a property field that advises the IP subnet supported by the Manager. If the Manager supports more than one IP subnet, then multiple registrations are carried out. The Manager object obtains knowledge of the subnets it supports via a properties file, controlled by the system manager, which is read by the TAF Manager during the registration process.

The TAF Manager CORBA object offers one method - initiate probe. It expects two parameters - the source IP address (ie the IP address of the image server in IMAD) and the destination IP address (ie the address of the end user). The method will return the minimum expected bandwidth, the likely expected bandwidth and the maximum expected bandwidth between the two hosts. An extract of the CORBA interface definition language (IDL) description of TAF is at Appendix One.

TAF Service clients (in the case of IMAD, these are the Image Query Managers, not the end user clients) can query any TAF Manager, but typically will use the most convenient TAF Manager (ignoring the IP subnet property). The initial action of the

TAF Manager is to identify the appropriate TAF Agent to probe the destination IP address. In the first instance the TAF Manager will consult its local list of subnets it supports to see if its organic TAF Agent should be employed. Otherwise, the TAF Manager queries the trader to locate a TAF Manager advertising support for an IP subnet corresponding to the source IP address. Assuming an appropriate TAF Manager is identified, then it is called by the initial TAF Manager as if the original TAF Manager was any other TAF Service client. The appropriate performance parameters will be passed back via this chain to the TAF Service client.

7. Conclusions and Further Work

A basic Transmission Availability Forecaster service has been developed for a best effort IPv4 network:

- Trials to date have shown it can successfully reduce the range of uncertainty of available capacity from perhaps 4000:1 to under 17:1 (and typically better than 4:1).
- The probe is intended to be used immediately before a download and so will take account of long term changes in the network situation.
- The probe measurement takes a finite period and so can aggregate and characterise short term variations in network situation.
- Nevertheless, the process remains vulnerable to medium term variations, ie changes that occur after the probe and before the download.

There are some constraints with the current implementation that ought to be addressed in the future:

- The current implementation is based on constant bit rate trunks between the routers. It is not yet known how well the prediction mechanism will operate in an environment where the routers implement non-constant capacity trunks.
- High speed connections where there is no clear bottleneck may prove to be problematic.
- While not relevant to ExC3ITE, DSTO ought to examine extensions to the TAF to cater for errored channels.
- The TAF service should address the situation where the client downloads multiple images simultaneously.

8. References

[1] M.W. Grigg et al, "Implementation of a Distributed Imagery Library System", submitted as DSTO Technical Report, 1999.

[2] G.R. Wright and W.R. Stevens, TCP/IP Illustrated Volume 1: The Protocols, Addison-Wesley, 1994, ISBN 0-201-63346-9.

[3] V. Jacobson, "traceroute", Network Research Group, Berkeley, CA
(Available at ftp://ftp.ee.lbl.gov/traceroute.tar.Z)

[4] V. Jacobson, "Pathchar - a tool to infer characteristics of Internet paths", Network Research Group, Berkeley, CA
(Available at http://www.caida.org/Pathchar/).

[5] A. S. Tanenbaum, *Computer Networks,* Third Edition, New Jersey: Prentice Hall International.

[6] V. Paxson, End-to-End Internet Packet Dynamics, *ACM SIGCOMM '97,* September 1997, Cannes, France
(Available at ftp://ftp.ee.lbl.gov/papers/vp-pkt-dyn-sigcomm97.ps.Z).

Appendix A: Additional Details

A.1 IP Probe Details

A.1.1 IP Payload Size Sweep Concept

Information is transmitted on the network in packets. Each protocol layer adds overheads. Figure 1-1 illustrates the structure of a packet carrying an ICMP payload. As the IP model was to provide data to the TCP model to assist in calculating effective data throughput to the TCP layer, the overheads included in T_{PQ} were only to include IP and any physical headers. The ICMP header is factored as part of the IP payload when calculating the bits transmitted.

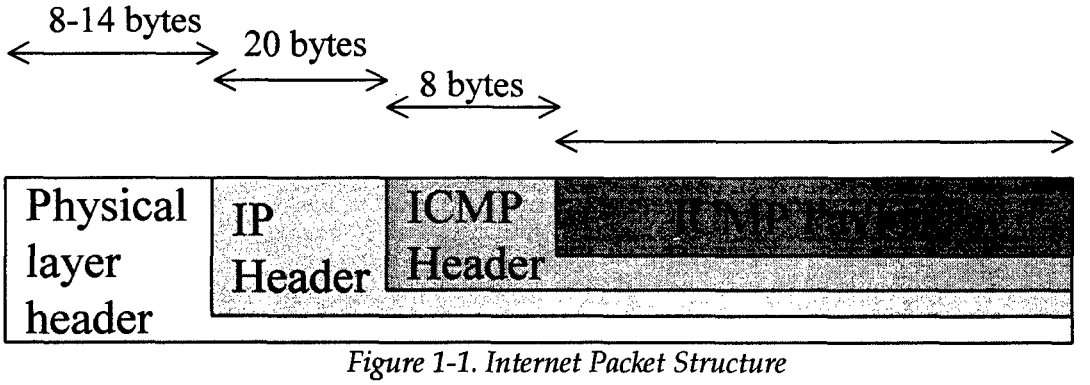


Figure 1-1. Internet Packet Structure

The initial approach taken in the IP probe development was to emulate the processing depicted in Figure 4 (in the main body of the report). A number of packets of different length would be sent - the change in OTT caused by change in payload size would determine the raw bandwidth, while the time taken for a packet of zero length would provide T_{PQ}.

Table 1-1 and Figure 1-2 show the results from probing action during a quiet period on the DSTO corporate network. This particular connection is passing from one LAN to another via two routers interconnected by a 128 kbps ISDN line. The results show that except for some non-linearities at very small packet sizes, the slope method gives a good estimation of the bottleneck bandwidth. The estimated bit rate from the slope is 115.8 kbps.

ICMP Payload(bytes)	*IP Payload (bytes)*	*Return Time (msec)*	*One Way Time (msec)*
10	18	58	29
20	28	60	30
32	40	62	31
64	72	66	33
128	136	75	37.5
256	264	93	46.5
512	520	128	64
1024	1032	207	103.5

Table 1-1. OTT for Various Packet Sizes

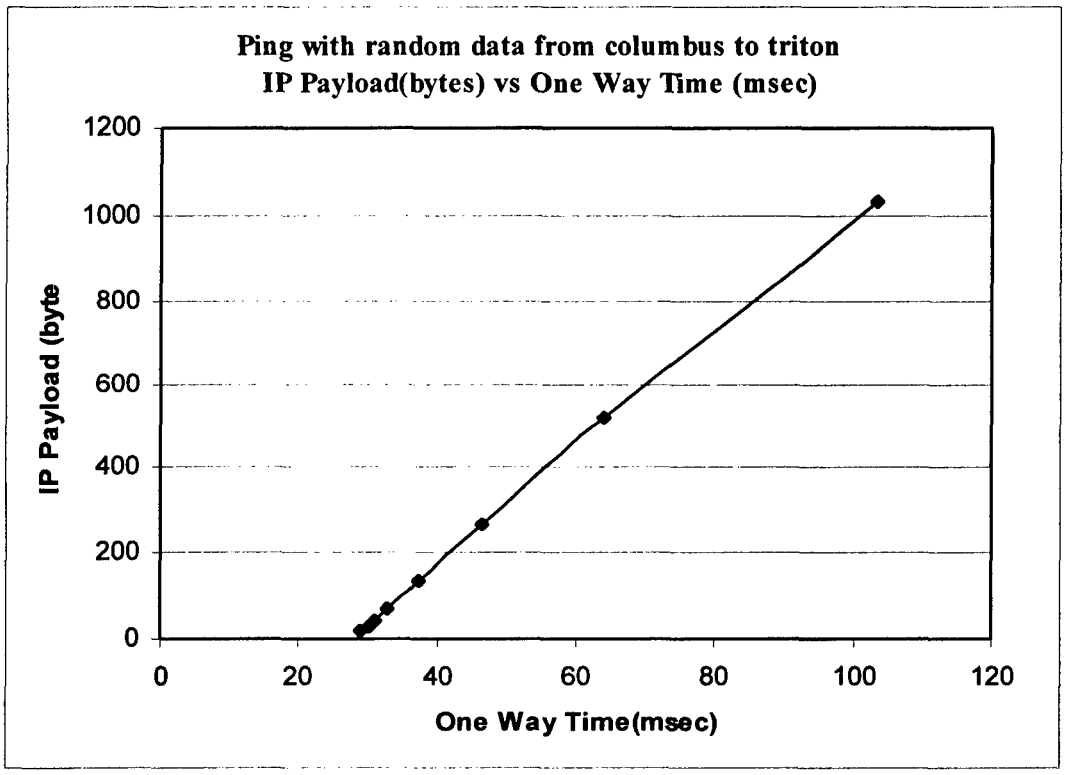


Figure 1-2. Graphical Depiction of Measured Performance

A.1.2 IP Impact of Network Traffic on Probe Measurements

Each sweep through a sequence of probes with different packet lengths gives one estimate of T_{PQ} (the extrapolated intercept on the time axis where packet length is zero). As stated earlier, one of our fundamental assumptions is that the raw bottleneck bit rate remains constant (variations in user capacity are caused by variations in the shared access to that bandwidth). A series of sweeps should in principle see constant slope but have variations in T_{PQ} reflecting variations in the number and size of packets

queued ahead of the probe. Linear regression was to be conducted on the measurements of each sweep (to overcome any minor variations in individual packet timings). The collection of time axis intersects provided a measure of the statistics of T_{PQ} for consideration in the TCP model method.

This method proved untenable. Figure 1-3 shows three sweeps through packet lengths of 18, 28, 40, 72, 136, 264, 520, 1032 and 2056 bytes. (Beware of the transposition of axes compared with Figure 1-2). One sweep (marked by diamonds) shows the performance one would expect with larger packets suffering proportionately larger delays. The other two sweeps (marked by squares and triangles) do not show this characteristics. Note for instance the sixth probe of the triangle sweep (264 bytes) which takes substantially less time than the fifth probe (136 bytes). Linear regression of these lines does not give constant slope. The reason for this anomaly is easily deduced. The assumption that the intercept will provide an estimate of T_{PQ} is based on the assumption that each of the probe packets face a similar queue ahead of them as other packets in the sweep. If a smaller probe packet encounters a large queue and the subsequent larger probe packet encounters only a small queue then the expected increase in RTT may not eventuate. Accordingly an alternative approach was developed.

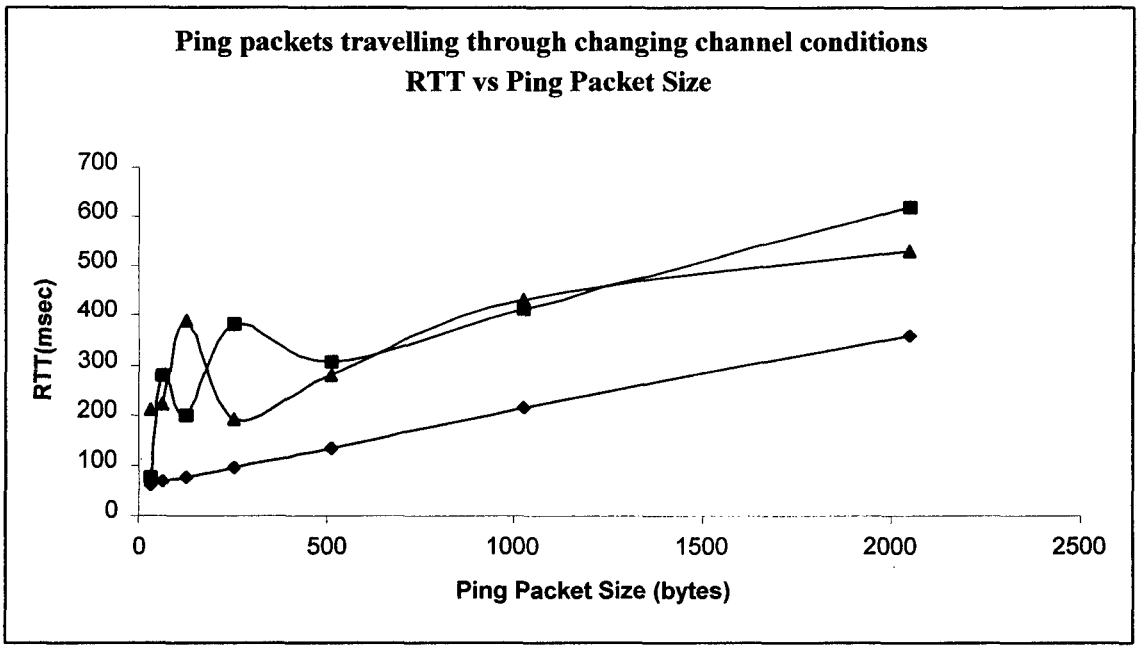
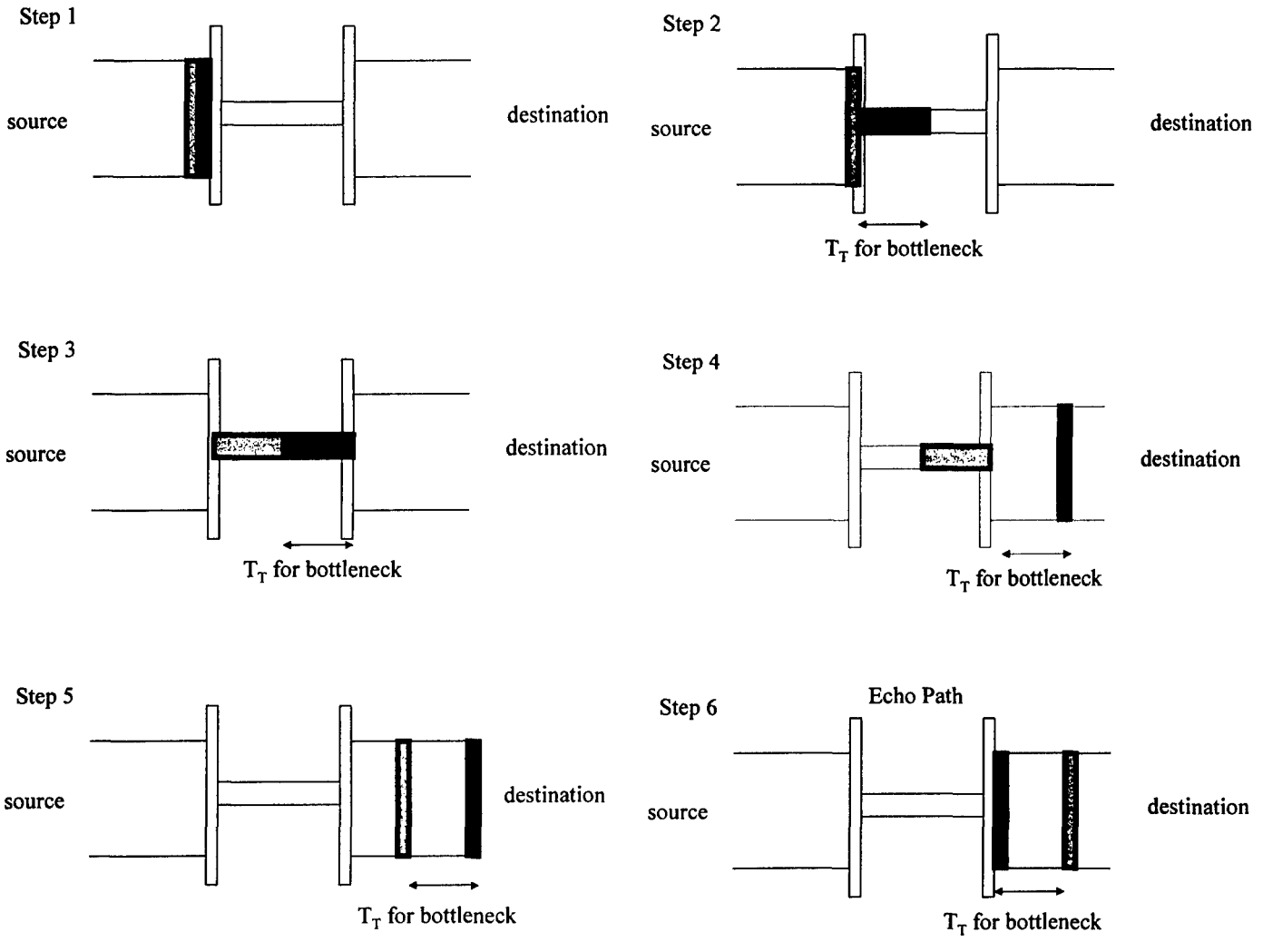


Figure 1-3. Illustration of ping packets experiencing stressed channel conditions

A.1.3 IP Packet Burst Concept

We are indebted to Paxson [6] and other references cited by him for introducing us to the concept of multiple packet probing to overcome the problems discussed in para 1.2. In effect the first packet in each burst acts to clear the outstanding queues. The second packet suffers a similar queuing delay to the first packet **plus the additional queuing delay from the need to clock the first probe packet through the bottleneck**. This means that probe packets arrive back at the source separated in time by the T_T of the preceding probe packet, as illustrated in Figure 1-4. There is an underlying assumption that there is no insertion of packets between consecutive probe packets and this situation must be identified during measurement activities.

In Figure 1-4 the width of the transmission pipes in the figure is indicative of the bit rate. The area of the shaded packets is indicative of the number of bits in the packet - thus the length of the packet in the pipe (horizontal direction) is indicative of the length of time the packet occupies the pipe and the relative timing of the packets.



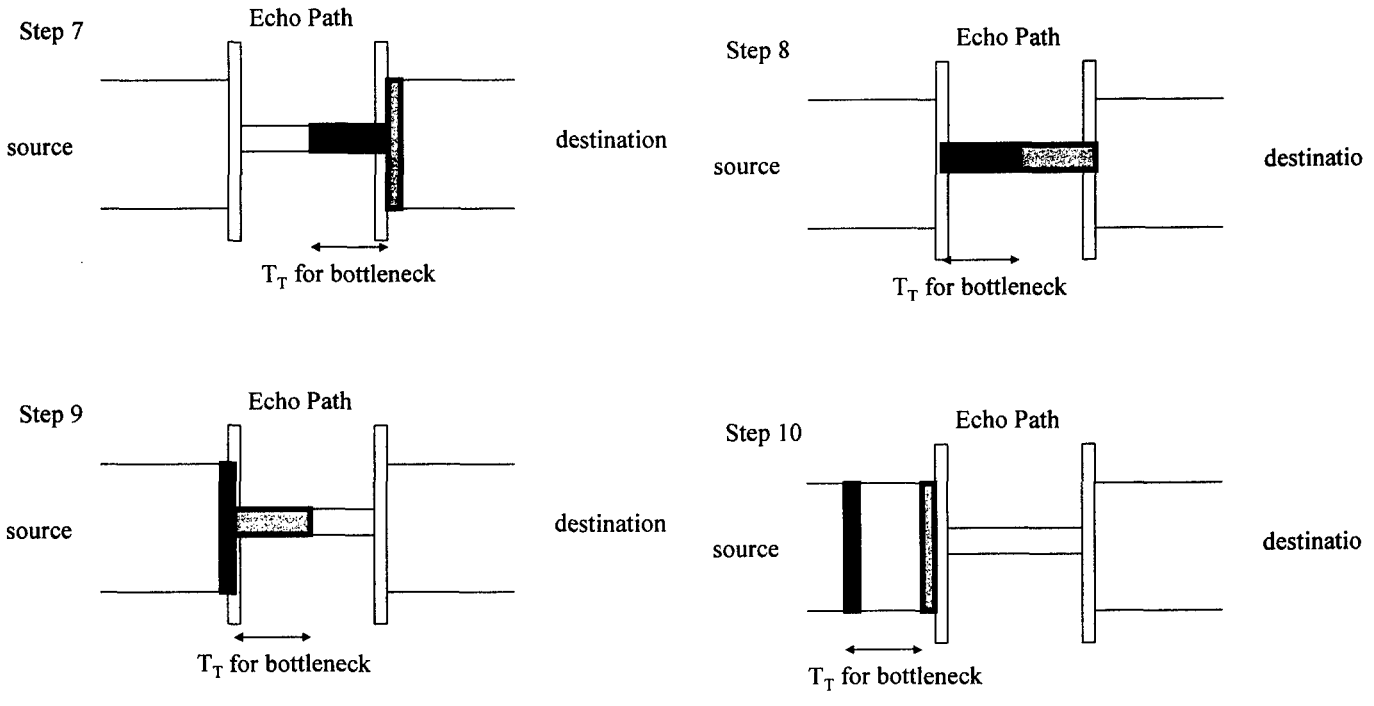


Figure 1-4. Deducing T_T Using Multiple Packet Probes.

This procedure thus provides a means of estimating T_T and hence T_{PQ} from the OTT.

$$OTT = T_{PQ} + T_T$$

Thus

$$T_{PQ} = OTT - T_T$$

or

$$T_{PQ} = \frac{RTT}{2} - T_T$$

A second benefit of this approach is that as a better mechanism for measuring T_T, it provides a better mechanism for estimating the bottleneck bandwidth. Multiple bursts of packets of different sizes can be used to measure the changes in T_T with changes in packet size and thus calculate the bottleneck bandwidth as per the sweep approach.

A.1.4 IP Impact of Network Traffic on Burst Probe Measurements

The burst packet technique described earlier assumes that no packets from other communications connections are inserted by the network to disturb the relative timing of the packets arriving back at the probe insertion point. If this were the case, a sequence of T_T estimates would provide a consistent measure (an example of an ideal measurement is given in Table 1-2. The calculation of the difference in T_T measurements is included to show that the T_T figures are consistent and will be used in later examples.

Packet	*OTT*	T_T	*Difference in* T_T
1	500		
2	510	10	
3	520	10	0
4	530	10	0
5	540	10	0
6	550	10	0
7	560	10	0
8	570	10	0

Table 1-2. Ideal Burst Performance

There are two key failure modes to the assumption that no packets will be inserted between probe packets:

- A packet might be inserted, typically at the bottleneck. This delays the next probe packet **and all subsequent packets**.
- A packet might be inserted, after the first passage through the bottleneck, delaying the next probe packet **but not by so long that subsequent packets catch up.**

The first situation is shown in table 1-3 where the non-probe packet has been inserted between packets 3 and 4. Probe packet 4 has suffered a delay of 60 (ie the typical T_T of 10 plus an additional delay of 50). The arrival time of packet 5 and those following has been similarly delayed thus restoring the typical inter-packet delays - the difference in T_T becomes momentarily negative. A running total of the difference in T_T shows that the packet 4 data has been disturbed.

Packet	*OTT*	T_T	*Difference in* T_T	*Running Total of Difference in* T_T
1	500			
2	510	10		
3	520	10	0	0
4	580	60	50	50
5	590	10	-50	0
6	600	10	0	0
7	610	10	0	0
8	620	10	0	0

Table 1-3. Insertion with All Subsequent Probe Packets Delayed

The second situation is shown in table 1-4 where the non-probe packet has been inserted between packets 3 and 4. Probe packet 4 has suffered a delay of 15 (ie the typical delay of 10 plus an additional delay of 5) but this delay is insufficient to cause packet 5 to catch up to packet 4. The arrival time of packet 5 is at the correct time (in absolute sense) as if packet 4 had progressed correctly. As a consequence, however, the T_T is artificially shortened by the increase in T_T for packet 4. A running total of the difference in T_T shows that the packet 4 data has been disturbed, by the inserted packet, and the data for packet 5 has been artificially disturbed.

Packet	*OTT*	T_T	*Difference in* T_T	*Running Total of Difference in* T_T
1	500			
2	510	10		
3	520	10	0	0
4	535	15	5	5
5	540	5	-10	-5
6	550	10	5	0
7	560	10	0	0
8	570	10	0	0

Table 1-4. Insertion with No Subsequent Probe Packets Delayed

A.1.5 IP Implementation of the Burst Probe Measurement

This experiment sent a series of ping packets to the destination. Two packet sizes were used, originally 128 and 512 bytes (later changed to 200 and 700 bytes). Each simulation run consisted of 12 ping packets 6 of each size (giving 4 estimates of T_T for each packet size and two T_{PQ} estimates). 100 such simulations were tried. A mechanism similar to that described above was employed to ignore spurious results. The implementation

applied a defined threshold of acceptance on the "Running Total of Difference in T_T " of 25% of an approximate T_T for that burst.

This trial gave median figures for T_T for 512 bytes and 128 bytes of 34.55 msec and 10.18 msec respectively. This corresponds to a bottleneck bit rate of 126 kbps a very close match to the link rate.

Figure 1-5 provides the histogram of T_T measured for 128 byte probes. Figure 1-6 provides the histogram of T_T measured for 512 byte probes. While Figure 1-7 provides the histogram of T_{PQ} measured. T_{PQ} was estimated to be 33msec. One can infer from this distribution that most of the packets did not experience any queuing delays. As the network gets more congested the characteristic peak spreads into larger delays.

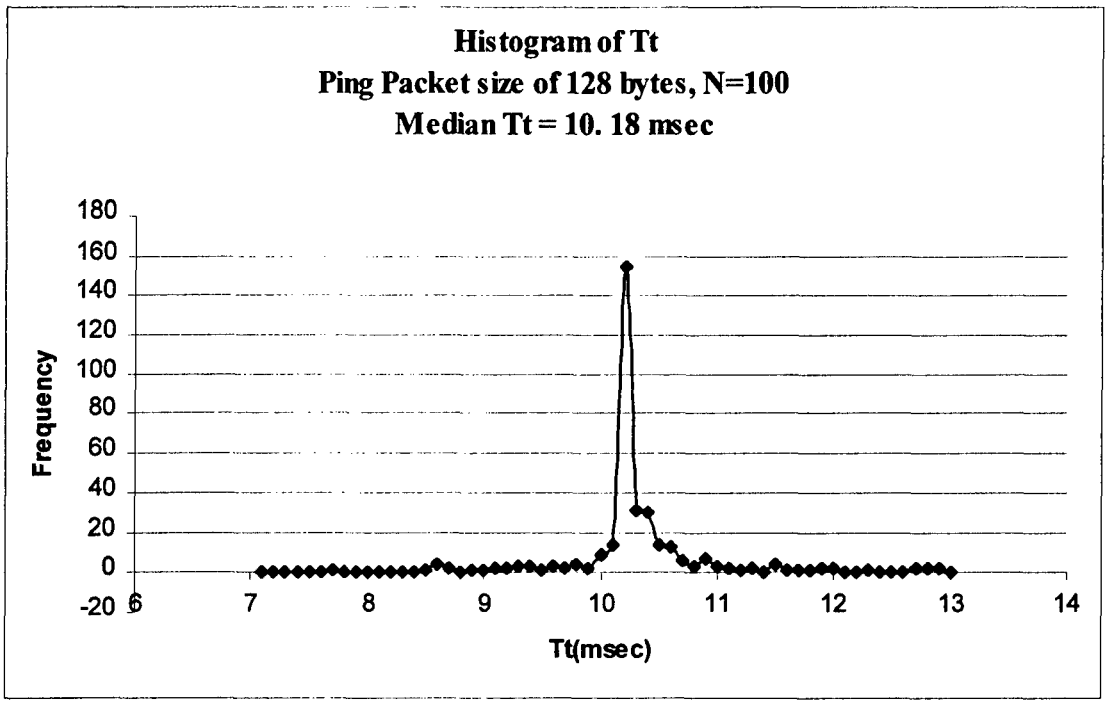


Figure 1-5 Typical distribution of T_T in msec for a packet size of 128 bytes

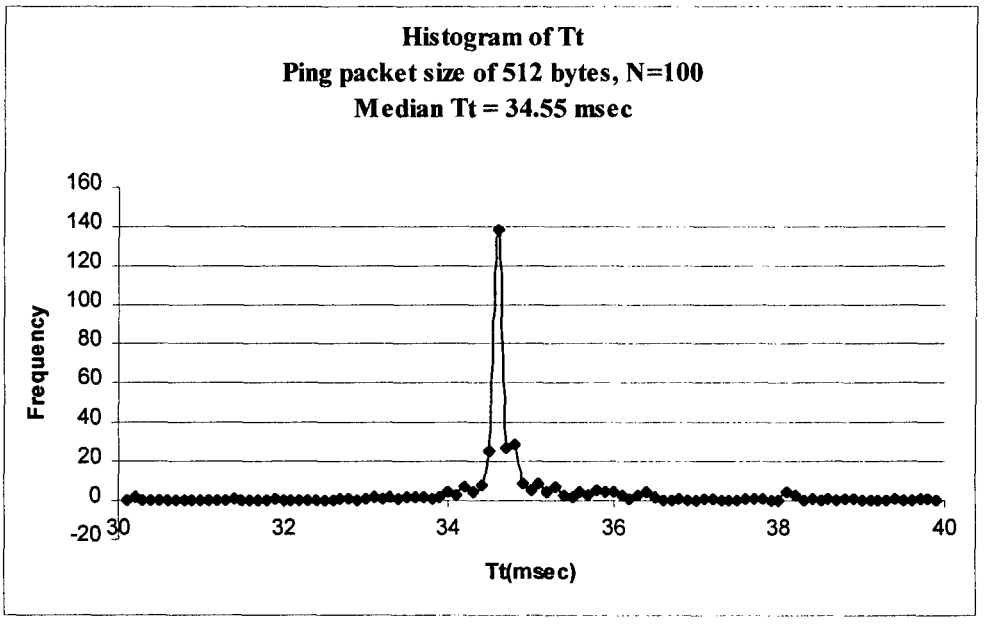


Figure 1-6 Typical distribution of T_T in msec for a packet size of 512 bytes

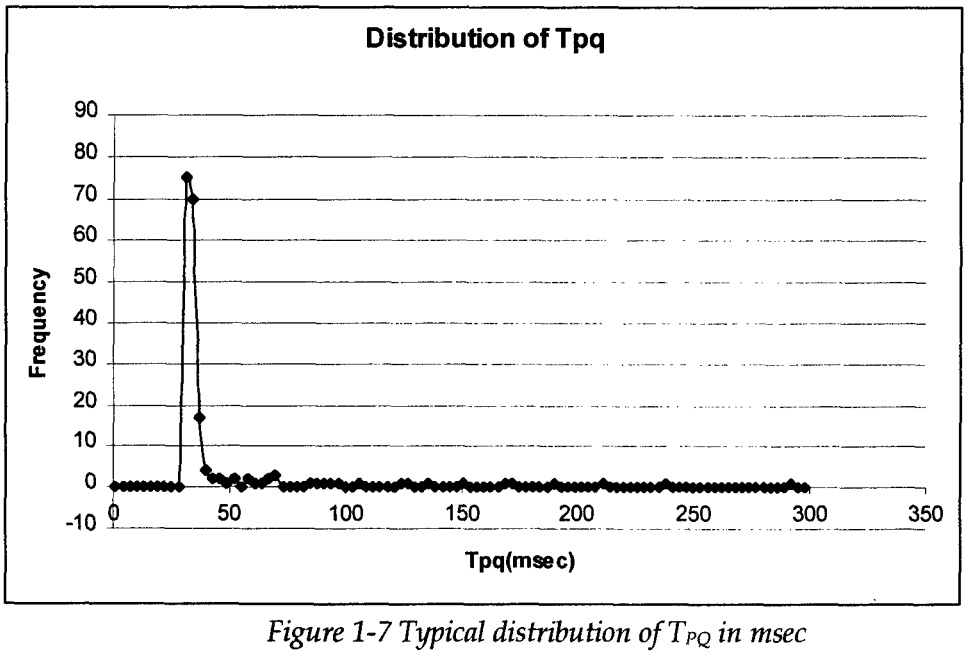


Figure 1-7 Typical distribution of T_{PQ} in msec

A.2 TCP Model Details - Normalised Estimated Bandwidth Concept

Trials of the probe program led on to the development of the TCP model and specifically the bandwidth estimation element. While the upper and lower bandwidth

bounds determination was straightforward (discussed in the report body), the estimator for the likely bandwidth required considerable examination using the IP parameters extracted from trials.

Trials of TAF attempted to encompass three dimensions: file size, distance (router hop count) and busyness of the network. Initial trials to develop the likely bandwidth estimator comprised twenty seven instances of probe and confirmatory download covering:

- three different file sizes: 3 Mbytes, 1 Mbytes and 10 kbytes,
- three different network connections:
  - local connection over the DSTO corporate network - three hops with a relatively low bandwidth connection,
  - national connection – nine hops over the Internet within Australia,
  - international connection – 23 hops over the Internet to the USA.

As noted earlier, the T_{PQ} estimate from the probe is not strictly packet length independent and this was seen in particularly stable trials. In principle the additional T_T elements (which can not be measured) ought to be increased in proportion to the increase from probe packet size to data packet size. Nevertheless, the T_{PQ} estimates are dominated by queuing delays on the multihop links. Therefore, we consider that the additional T_T elements don't impact greatly on the maximum T_{PQ} figure used in the bounding calculations.

We anticipated that TAF estimations for the 10 kbytes files would be less accurate. Our reasoning was that: first, such files were too short for TCP to stabilise into some kind of steady state even if the network was stable. Second, the short time taken for the download would mean that the short term variations in network availability would have a more significant impact on the effective throughput.

The trials validated this expectation with two (out of nine) of the 10 kbyte downloads experiencing performance less than the minimum predicted bit rate. Of less concern given our goal of pessimistic estimates, one download experienced performance exceeding the predicted maximum. By contrast, the 1 Mbytes and 3 Mbytes file downloads were better behaved with the experienced download performance exclusively falling in the bounds. As our intended application will primarily be dealing with large file sizes, this result was encouraging.

While the success in bounding the actual performance was encouraging, the range of the bound from high to low was up to 17:1. This offers some utility to the client, but better accuracy was sought. In continuing this work, a measure entitled *Normalised Expected Bit rate* (NEB) was defined. The NEB is a linear measure giving a figure between 0 and 1 indicating where the achieved bit rate occurred in the range of minimum expected bit rate (where the NEB would be 0) to maximum expected bit rate (where the NEB would be 1).

While there is some (inverse) correlation between the number of timeouts experienced by the probe process and the NEB achieved by the download, a consistent mathematical relationship could not be determined. This was especially noticeable in the case of the local trials where variations in NEB were experienced without any timeouts at all.

Instead, consideration was given to the relationship between NEB and a factor derived from the mean T_{PQ} and the bottleneck bit rate. The product of the mean T_{PQ} and the bottleneck bit rate is indicative of the mean number of bits in the bottleneck queue. The NEB was found to be quite strongly (inversely) correlated with this figure so it was used as the basis of estimating the likely bit rate in the production TAF service. Trials were then conducted to determine a suitable formula for predicting NEB. The formula:

$$\frac{5000}{MeanT_{PQ} * BottleneckBitRate}$$

was found to provide a conservative NEB predictor and was the basis of subsequent analysis.

A.3 The TAF CORBA Interface

This extract of the TAF object CORBA interface definition language (idl) describes the methods offered by TAF and the data interchanged between TAF clients and the TAF service.

```
module PackageTAF{

  interface TAFMgr;

  typedef string IPaddress;
  enum NodeName
  {
    DSTO_FHP, DSTO_SALISBURY, DSTO_HQAST, DJFHQ
  };

  struct ChannelCharacteristics
  {
    IPaddress source_address;
    NodeName source_nodename;
    IPaddress destination_address;
    NodeName destination_nodename;
    unsigned long maximum_throughput;
    unsigned long likely_available_throughput;
```

```
    unsigned long minimum_throughput;
  };

  typedef sequence <ChannelCharacteristics> ChannelList;
  struct ExceptionInfo {
    string exception_details;
  };
  exception NoChannelAvailable {ExceptionInfo info;};

   interface TAFMgr // Tx Availability Forecast Manager
  {

    ChannelCharacteristics probe_channel (
      in IPaddress source_address,
      in IPaddress destination_address )
    raises (NoChannelAvailable);

  };
}; // End of module Package TAF
```

DISTRIBUTION LIST

A Transmission Availability Forecast Service for Internet Protocol Networks

W.D. Blair and R. Jana

AUSTRALIA

DEFENCE ORGANISATION

Task Sponsor **DGC3ID**

S&T Program

Chief Defence Scientist
FAS Science Policy } shared copy
AS Science Corporate Management
Director General Science Policy Development
Counsellor Defence Science, London (Doc Data Sheet)
Counsellor Defence Science, Washington (Doc Data Sheet)
Scientific Adviser to MRDC Thailand (Doc Data Sheet)
Scientific Adviser Policy and Command
Navy Scientific Adviser (Doc Data Sheet and distribution list only)
Scientific Adviser - Army (Doc Data Sheet and distribution list only)
Air Force Scientific Adviser
Director Trials

Aeronautical and Maritime Research Laboratory
Director

Electronics and Surveillance Research Laboratory
Director (1 copy)

Chief of Communications Division
Research Leader Military Information Networks
Head Network Integration Group
Author(s):
W.D. Blair
R. Jana

DSTO Library
Library Fishermens Bend
Library Maribyrnong
Library Salisbury (2 copies)
Australian Archives
Library, MOD, Pyrmont (Doc Data sheet only)
*US Defense Technical Information Center, 2 copies
*UK Defence Research Information Centre, 2 copies
*Canada Defence Scientific Information Service, 1 copy
*NZ Defence Information Centre, 1 copy
National Library of Australia, 1 copy

Capability Development Division

Director General Maritime Development (Doc Data Sheet only)
Director General Land Development (Doc Data Sheet only)

Navy

SO (Science), Director of Naval Warfare, Maritime Headquarters Annex, Garden Island, NSW 2000 (Doc Data Sheet and distribution list only)

Army

ABCA Office, G-1-34, Russell Offices, Canberra (4 copies)
SO (Science), DJFHQ(L), MILPO Enoggera, Queensland 4051 (Doc Data Sheet only)
NAPOC QWG Engineer NBCD c/- DENGRS-A, HQ Engineer Centre Liverpool Military Area, NSW 2174 (Doc Data Sheet only)

Intelligence Program

DGSTA Defence Intelligence Organisation

Acquisitions Program

Director General Command & Support Systems, Electronic Systems Acquisition Division

Corporate Information Program

DISC-Deakin

Corporate Support Program (libraries)

OIC TRS, Defence Regional Library, Canberra

UNIVERSITIES AND COLLEGES

Australian Defence Force Academy
Library
Head of Aerospace and Mechanical Engineering
Deakin University, Serials Section (M list), Library, Geelong, 3217 Senior Librarian, Hargrave Library, Monash University
Librarian, Flinders University

OTHER ORGANISATIONS

NASA (Canberra)
AGPS
State Library of South Australia
Parliamentary Library, South Australia

OUTSIDE AUSTRALIA

ABSTRACTING AND INFORMATION ORGANISATIONS

INSPEC: Acquisitions Section Institution of Electrical Engineers
Library, Chemical Abstracts Reference Service
Engineering Societies Library, US

Materials Information, Cambridge Scientific Abstracts, US
Documents Librarian, The Center for Research Libraries, US

INFORMATION EXCHANGE AGREEMENT PARTNERS

Acquisitions Unit, Science Reference and Information Service, UK
Library - Exchange Desk, National Institute of Standards and Technology, US

SPARES (5 copies)

Total number of copies: 54

DEFENCE SCIENCE AND TECHNOLOGY ORGANISATION
DOCUMENT CONTROL DATA

1. PRIVACY MARKING/CAVEAT (OF DOCUMENT)

2. TITLE

A Transmission Availability Forecast Service for Internet Protocol Networks.

3. SECURITY CLASSIFICATION (FOR UNCLASSIFIED REPORTS THAT ARE LIMITED RELEASE USE (L) NEXT TO DOCUMENT CLASSIFICATION)

Document	(U)
Title	(U)
Abstract	(U)

4. AUTHOR(S)

W.D. Blair and R. Jana

5. CORPORATE AUTHOR

Electronics and Surveillance Research Laboratory
PO Box 1500
Salisbury SA 5108 Australia

6a. DSTO NUMBER	6b. AR NUMBER	6c. TYPE OF REPORT	7. DOCUMENT DATE
DSTO-RR-0146	AR-010-844	Research Report	December 1999

8. FILE NUMBER	9. TASK NUMBER	10. TASK SPONSOR	11. NO. OF PAGES	12. NO. OF REFERENCES
E8709/4/12(2)	JNT96/295	DGC3ID	26	6

13. DOWNGRADING/DELIMITING INSTRUCTIONS

14. RELEASE AUTHORITY

Chief, Communications Division

15. SECONDARY RELEASE STATEMENT OF THIS DOCUMENT

Approved for public release

OVERSEAS ENQUIRIES OUTSIDE STATED LIMITATIONS SHOULD BE REFERRED THROUGH DOCUMENT EXCHANGE CENTRE, DIS NETWORK OFFICE, DEPT OF DEFENCE, CAMPBELL PARK OFFICES, CANBERRA ACT 2600

16. DELIBERATE ANNOUNCEMENT

No Limitations

17. CASUAL ANNOUNCEMENT Yes

18. DEFTEST DESCRIPTORS

Internet, TCP/IP (Computer network protocol), Bandwidth, Communications traffic, Data transmission, CORBA (Computer architecture), Distributed computer systems

19. ABSTRACT

The quality of available network connections, especially the bandwidth available to clients using the connections, has a large impact on the performance of distributed applications. For example, document transfer applications such as the World Wide Web suffer a dramatic increase in response times as a result of network congestion causing a reduction in the available bandwidth of the connection. This paper recognises the explosion of interest in the use of Internet Protocol (IP) networks within the Australian Defence Organisation and describes the development of a software tool for estimating the available bandwidth between a server and client in a distributed computing environment. We discuss the design and implementation details of the Transmission Availability Forecaster (TAF) probe and present validation studies demonstrating its reliability and accuracy in the context of actual Internet conditions.