Passive Calibration of an Active Measurement System

Stephen Donnelly, Ian Graham, René Wilhelm

Abstract— **Measuring QoS in the internet today is difficult as notions of what constitutes QoS vary. Service Level Agreements between customers and network service providers are often poorly defined. The most workable SLAs currently are defined in terms of how long packets may take to travel from some host to another host, and what acceptable levels of packet loss are.**

This paper presents a direct comparison of active One-way-Delay measurements with an independent passive measurement system over an intercontinental distance.

A high level of agreement was found between the active and passive systems in this study. It is hoped that this work will help researchers making active measurements to appreciate error contributions from end-point equipment when interpreting their results.

Keywords— **Passive measurement, Active measurement, Calibration, RIPE, Test Traffic**

I. Introduction

Measuring the time taken for packets sent from one host to arrive at their destination host requires the ability to detect these packets at both the source and destination, and to record time-stamps from a common clock for each observation. The accuracy of any system designed to perform this measurement will depend on its two necessary components, a common clock, distributed to the vicinity of source and destination host, and a mechanism to observe the packets, and accurately attach a time-stamp to each observation from the distributed clock.

There are several mechanisms capable of providing a distributed clock. NTP is a software only approach, which calibrates a host's clock to a server by exchanging packets over a connecting network, and estimating the delay to and from the server[1]. A common server could be used by both end-points, but more commonly separate servers would be used, with each server depending on some external clock. This approach is however susceptible to changing or asymmetric network delays, and so its accuracy is typically not expected to be better than plus or minus 1ms offset from the time standard.

If higher accuracy is desired, an alternative to a network based clock synchronisation method must be used, such as providing a reference clock directly to each end-point of the measurement. This requires extra hardware, such as Global Positioning System (GPS) or CDMA receiver, and may require the mounting of an antenna on the exterior of the building. Although expensive and potentially difficult to install, such systems promise accuracies of plus or minus 100ns or better to UTC [2].

Once a clock is available, a means of observing the packets and time-stamping them with the clock is needed. The simplest method requiring no additional hardware to the host is to use NTP either from a server or from a local reference clock to condition the kernel clock of the host, and then use libpcap to time-stamp the packets as received by a conventional Network Interface Card (NIC). This may be acceptable where a NIC is available for the desired network media, but extra uncertainty in the time-stamps is added via the conditioning of the kernel clock, the NIC behaviour in reporting packet arrivals, interrupt latency of the host, and variations in processing of the packet arrival by the network stack.

A formal metric for One-way-Delay is defined by the IETF IPPM working group in RFC2679, "A One-way Delay Metric for IPPM" [3]. This document defines a singleton metric called *Type-P-One-way-Delay* that describes a single measurement of One-way-Delay. Using this singleton metric, a sample metric is defined which refers to a sequence of Type-P-One-way-Delay singleton delays measured at times taken from a Poisson process, called *Type-P-One-way-Delay-Poisson-Stream*.

There are at least two implementations of measurement systems to collect these metrics, by ANS[4], and by RIPE NCC[5].

In this paper, we investigate the use of GPS synchronised hardware packet capture technology to calibrate the RIPE NCC Test Traffic Measurement active measurement system in its operational setting.

II. Test Traffic Measurement System

Since 1997, Réseaux IP Européens Network Coordination Centre (RIPE NCC) has been operating a system called Test Traffic Measurements (TTM) for measuring One-way-Delay[5]. TTM is an active measurement system which has implemented the IPPM One-way Delay and One-way Loss metrics to perform independent measurements of connectivity parameters in the Internet. After a succesfull development and pilot phase, TTM became a regular RIPE NCC membership service in October 2000.

In TTM active probe packets containing time-stamps are sent from a dedicated measurement PC running FreeBSD on the source network to a similar PC on the destination network. The TTM System is illustrated in Figure 1. Independent probe packets are sent from the TTM system on the second network to the first to measure the One-way-Delay in the return direction.

Stephen Donnelly is a Ph.D. student at The University of Waikato, Hamilton, New Zealand, where Professor Ian Graham is the Dean of the School of Computing and Mathematical Sciences. Email sfd@cs.waikato.ac.nz or ian@waikato.ac.nz

René Wilhelm works for RIPE NCC, Amsterdam, The Netherlands. Email wilhelm@ripe.net

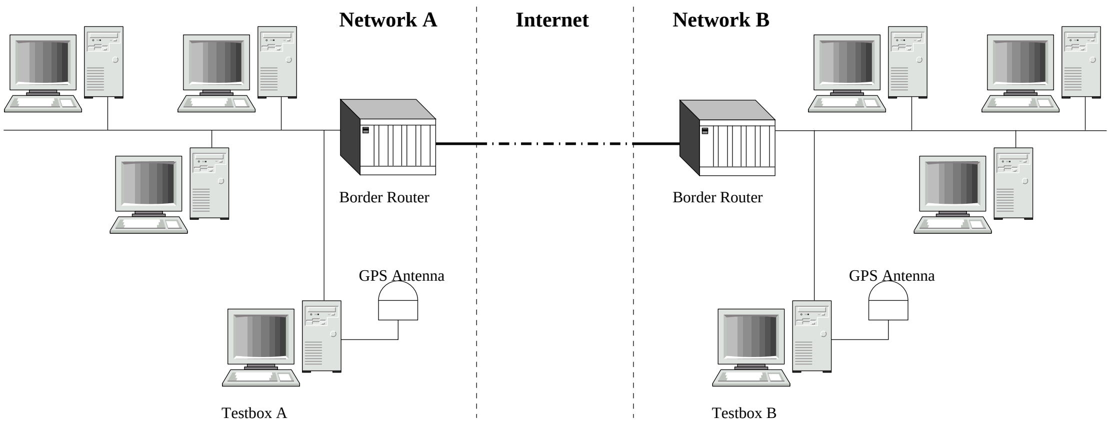

Fig. 1. RIPE NCC Test Traffic Measurement System

The probe packets are 128 bytes long, and contain a UDP frame with destination port 6000, and a UDP payload of 100 bytes. This is the TTM systems Type-P definition for the Type-P-One-way-Delay metric framework.

Each TTM PC is equipped with a GPS receiver. This receiver outputs a short pulse every second on the DCD control line of the PCs serial port. These pulses generate interrupts, which are used by NTP to phase-lock the kernel system clock.

A user-space daemon on the TTM PC schedules test probes. When a probe is to be sent, the packet is built in user-space, and the kernel system clock is read. The time is inserted into the packet, and it is passed into kernel-space to be queued for transmission.

After the probe packet is received by the destination TTM machine, it is time-stamped again with the kernel clock, which is synchronised by the same GPS mechanism. The difference between the time recorded in the packet, and the time of the packet arrival are subtracted to produce a One-way-Delay measurement.

Data collected at the TTM machines is later retrieved to a central point. By comparing the records from the source for packets sent, and from the destination for packets received, One-way-Loss can also be determined.

The user-space process on the source machine schedules probe packet transmissions. These are sent at a low rate in order to avoid artificially loading the network, especially where the number of TTM machines is large.

III. Dag Packet Capture System

The WAND group at the University of Waikato has developed the Dag monitor[6], a hardware and software system that passively receives and time-stamps network traffic in hardware. A block diagram of the Dag 3 series card is shown in figure 2. It consists of a network interface attached to a Field Programmable Gate Array (FPGA). The FPGA is also connected to the PCI bus, for transferring packet records to the host computer. A 32 bit RISC CPU shares a bus with the FPGA, and some memory. The host CPU can re-program the FPGA, which it uses to communicate with the host, and can also examine the packets as they are captured by the FPGA.

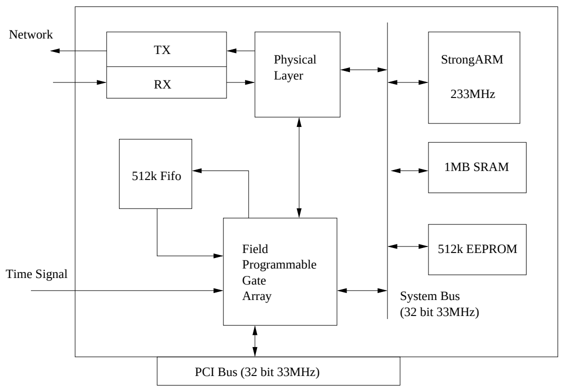

Fig. 2. Dag hardware measurement card

The Dag 3 series cards use a nominally 32MHz crystal oscillator as the basis for its time-stamping clock. These crystal oscillators however vary in frequency by up to 100 parts per million over time, and are temperature sensitive. In order to compensate for this drift, and to ensure the time-stamps produced are globally absolute with respect to UTC, a reference clock input is provided. Inside the FPGA, the 32MHz crystal oscillator clock is passed through a Direct Digital Synthesiser (DDS) to produce a clock running at 2^{24}Hz. The ratio between this desired frequency and the current actual frequency of the crystal oscillator is the control input to the DDS. The reference clock is compared to the synthesis clock every second, and any accumulated error is calculated. The control input to the DDS is then adjusted to null the error out over the next second. Once synchronised, this system will maintain the synthesised clock to within plus or minus two ticks of the reference signal, or approximately 120ns.

The GPS receivers used as reference clocks in this experiment claim to maintain synchronisation to UTC to plus to minus 100ns. In back to back tests of 2 systems with separate GPS receivers, the total difference between the synthesised clocks is less than plus or minus 500ns.

The FPGA latches its clock on the next rising edge of the synthesised clock after the physical layer indicates a valid frame has been received. This avoids the primary errors caused in NIC based systems, buffering on the NIC before the host is interrupted, and host interrupt latency.

Further information on the Dag series of measurement cards can be found on the Dag website, at http://dag.cs.waikato.ac.nz/.

IV. Calibration Methodology

Two computers containing Dag Ethernet monitor cards were installed at two RIPE NCC test-box sites. These systems passively collect and time-stamp the test-box active probes, providing an independent measure of the One-way-Delay of the packets from the time they leave one test-box until they arrive at their destination. The passive measurements at each site provide a very accurate measure of the difference between the test-box source time-stamp, and the time at which the packet actually appears on the source network. The difference between the time the packet arrives at the destination network and the test-boxes time-stamp of the event can be similarly found. Finally the passive time-stamps are compared to produce a 'wire One-way-Delay' measurement which is compared to the corresponding test-box One-way-Delay result.

The following sections use data from an experiment conducted from 00:00 UTC on the 12th of October 2000 to 00:00 UTC on the 13th. The specification of each of the TTMtest-boxes is different, both in the hardware, and the version of FreeBSD used. The details of the TTM machines used in this experiment are detailed in Table I. The GPS reference clock design is currently uniform across the deployed machines.

TABLE I
TTM hosts used in calibration experiment.

Host	*Site*	*FreeBSD*	*CPU*
tt01	Holland	2.2.8	Pentium 200MHz
tt47	New Zealand	2.2.1	Pentium II 233MHz

In order to obtain an independent measure of the One-way-Delay between tt01 and tt47, Dag measurement systems were introduced onto the Ethernet segments used by the measurement interfaces of each of the machines. This was done by attaching the Dag card's network interfaces to the same Ethernet hub that the TTM test-boxes used.

In each case the additional system was comprised of a PC containing Dag 3.1E passive Ethernet measurement card, and a separate GPS receiver. The GPS receiver connects directly to the network measurement card to control its local time-stamping clock. One interface from the Dag card is connected to a hub, along with the main Ethernet connection for the measurement machine, and the TTM host.

The Dag 3.1E at the time of the experiment was limited by firmware in the amount of data it could capture from each packet. The Dag 3.1E produces a 64 byte fixed length for each Ethernet frame that is received, regardless of the frames actual length [7]. When the per-record overhead (8 byte time-stamp plus 2 byte wire length) is subtracted from the 64 byte record, 54 bytes remain. This contains the first 54 bytes of the Ethernet frame, once the preamble has been discarded. The DIX Ethernet II frame format, used at both endpoints, defines a 14 byte header [8], including source and destination Media Access Controller (MAC) addresses, and a type field. The Ethernet payload fills the remaining 40 bytes of the record. An IPv4 header with no options requires 20 bytes. The Test Traffic probe packets are carried by the UDP protocol, which has an 8 byte header. This leaves only 12 bytes of UDP payload in the Dag record.

The Test Traffic probe packets observed have an IP total length of 128 bytes, or a UDP payload of 100 bytes. This means that not all of the probe packet payload is contained within the record returned by the Dag card. The Test Traffic probe packet format specifies that the sequence and time-stamp information carried within the packet be placed at a random byte offset within the UDP payload, and that an offset field should be at the start of the UDP payload in order to locate it.

This is done to try to make the packets less compressible, as this may affect the delay results for network paths were one or more link is compressed at the hardware layer for transportation. The consequence is that the TTM sequence number and time-stamp are not captured by the limited length Dag record.

In order to record these important fields, it is necessary to simultaneously capture the TTM probes with a system that can record the entire packet, even if its timing is less accurate. Since the main Ethernet card for the Dag host is connected to the measurement hub along with the Dag, it is sufficient to run `tcpdump` with a suitable 'snaplen' set in order to record the entire payload.

Over the experiment's 24 hour period, the Dag cards captured all Ethernet frames to host memory, and software on the host filtered these for packets of Type-P. The `tcpdump` run simultaneously on each Dag host used identical filter rules.

A. Transmission Latency

In the 24 hour experiment, tt01 transmitted a total of 58,498 probes to the various destinations, and tt47 transmitted 58,500 probes. As each probe packet is transmitted by the TTM machine, it is time-stamped by the Dag system, and the packet contents are collected by `tcpdump`. The time-stamp placed into the probe packet by the TTM system to represent the time that the probe was sent is extracted, and compared to the time-stamp recorded by the Dag hardware.

In post processing of the data recorded by the Dag card and by `tcpdump`, the packet trace data from the Dag card and from `tcpdump` are compared by IP source and destination addresses, as well as Id field in order to find matches. The TTM probe packet fields are then extracted from the `tcpdump` record, and the time-stamp from the corresponding Dag record. The difference between these time-stamps is the difference between when the TTM System recorded the packet as sent, and the time that the packet actually left the source TTM host. This difference is referred to below as the *transmission latency*.

In the specification of Type-P-One-way-Delay, the One-way-Delay is defined as the time taken between the probe packet has completely left the interface of the source machine, the *wire exit time*, to when the probe has completely arrived at the destination network interface, the *wire-arrival-time*. This means that any time difference between the recorded transmission time-stamp and the wire exit time is treated as an error. The upper bound on this error is referred to as *Hsource*. Likewise *Hdest* refers to an upper bound on the difference between the wire time of the packet at the destination and the destination time-stamp.

Since the Dag card records its time-stamp in hardware at the end of the preamble, and the packets are of fixed length, the Dag time-stamp is equal to the wire exit time of the probe packet, minus a offset due to the packet length. This varies depending on the line rate of the Ethernet. For 10Mbps Ethernet, the time required to send the 128 byte packet will be 102.4μs, for 100Mbps Ethernet it will be only 10.24μs. This allows us to use the the Dag card to measure the wire exit time of packets, and hence find an estimate for *Hsource*.

Figure 3 shows a time-series of transmission latencies per packet on a log scale. Although peak latencies surpass 10ms, the bulk of the samples fall below 200μs. The spikes in transmission latency are not outliers, they are actual measured latencies. There are a number of possible causes of these spikes, including cross traffic on the hub, and scheduling effects on the TTM host.

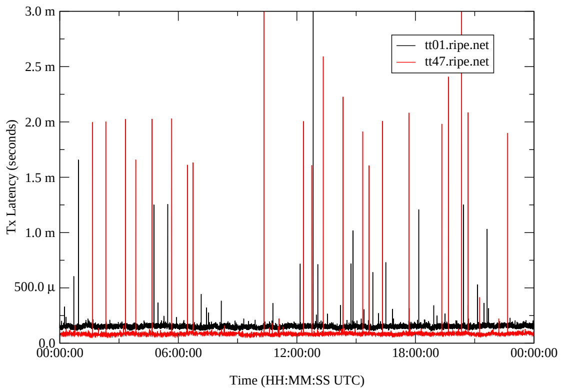

Fig. 3. TTM Transmission Latency

A histogram of transmission latency between 0 and 250μs is provided in Figure 4 in order to show the compact distribution of the bulk of the transmission latencies measured.

The TTM host tt01 has a transmission latency centred about 155μs, while tt47 is centred about 85μs. The offset between the two distributions is easily explained by the difference in processing power between the hosts. Differences in the Central Processing Unit (CPU) bridge and Peripheral Component Interconnect (PCI) bridge may also contribute, since the hosts use different motherboards with different chip-sets.

The 25 to 50μs spread of the bulk of the distribution may be caused be several factors, including variations in host interrupt latency, the time to process the packet, or

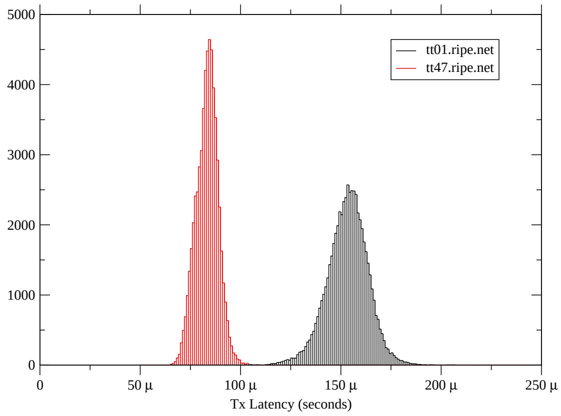

Fig. 4. TTM Transmission Latency distribution

NIC behaviour. Some NIC cards for instance will buffer received packets before delivering them to the host for up to 100μs, in order to reduce the interrupt load on the host in times of high network activty.

A cumulative histogram of the transmit latency focusing on the extreme values is presented in Figure 5. In this figure, the latencies of the two hosts have been normalised to each other by subtracting from each the median latency. This is justified as the measured latency is assumed to consist of some fixed offset plus some varying component, and we wish to compare the varying components. Host tt01 has a latency below 1ms in 99.98% of measurements, and below 250μs in 99.97%. Host tt47 has 99.96% of points below 1ms. 99.95% of measured latencies fall below 250μs for this host. The maximum latencies were 6.30ms and 11.93ms for hosts tt01 and tt47 respectively.

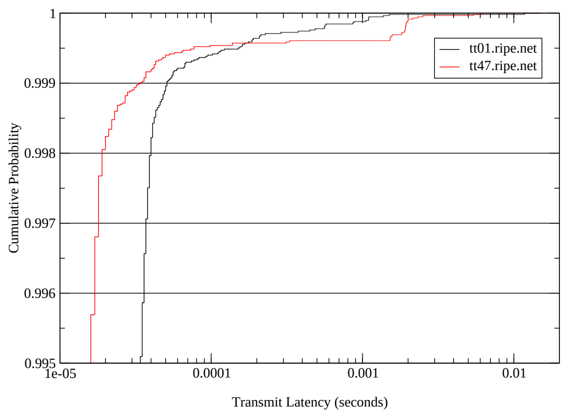

Fig. 5. TTM Transmission Latency cumulative distribution

Since a maximum length Ethernet frame takes 1.23ms on a 10Mbit Ethernet, spikes less than or equal to this value may be caused by the Ethernet NIC waiting for an existing packet on the hub to end before it can transmit. Transmission latencies greater than this value must be at least partially due to some other mechanism. Possibilities include the TTM host processor scheduling some other task in between the packet being time-stamped in user-space and the packet being sent to the NIC for transmission, or unusually long interrupt latencies due to slow peripheral devices such as hard disk controllers.

It may be possible to reduce the magnitude of latency peaks by rewriting the application so that it runs partly in kernel-space. A loadable kernel module could be called to insert the time-stamp into the packet, and queue it for transmission. The advantage is that kernel-space code is not pre-emptable, so scheduling concerns and context switches are avoided. In order to make the time-stamp to transmission operation atomic, it may be necessary to mask or disable interrupt processing during this operation. This approach cannot solve latency due to inaccuracies in the clock however, or cross traffic on the Ethernet media.

B. Transmission scheduling

Each TTM host transmits probe packets to 26 other TTM hosts, and each individual measurement is an example of the singleton Type-P-One-way-Delay metric. RFC2679 specifies only one sample metric for a sequence of singleton measurements, Type-P-One-way-Delay-Poisson-Stream. In this metric, the probe packets to a specific destination host should be sent at times taken from a Poisson process. This is referred to as Poisson sampling.

There are several advantages to using a Poisson process for scheduling measurements [9]. Poisson sampling of a system does not lead to synchronisation. That is, each sample is not predictable, so probes injected into the network cannot induce periodic behaviour. Periodic sampling can fail to measure periodic behaviour in the network, where the periods of the behaviour and the sampling are similar, or one is a multiple of the other. Poisson processes can also be proven to be asymptotically unbiased, even if the measurement process affects the network.

In order to implement Poisson sampling, the delays between sending probes should be independent, and random with a common exponential distribution with rate λ, equation 1.

$$G(t) = \lambda e^{-\lambda t} \qquad (1)$$

In 24 hours, the TTM host sends approximately 2160 probe packets to each of the destination TTM machines, or one every 40 seconds on average. Figure 6 shows the inter-transmission times for the first 500 probes sent from tt01 to tt47 and from tt47 to tt01.

Figure 7 shows the distribution of the observed probe spacing between 0 and 70 seconds. Overlayed is a Poisson distribution with lambda equal to 40 seconds, and an exponential distribution with rate 1/40. It is clear that the TTM system is not performing Poisson sampling, but rather sampling distributed in some approximately Normal way about a mean of 40 seconds. This distribution has been chosen for TTM because exponential distributions can on occasion generate very long times between probes, a problem not addressed in the specification of Type-P-One-way-Delay-Poisson-Stream.

C. Reception Latency

In the experiment as each packet from a remote TTM host arrives at its destination network, it is simultaneously

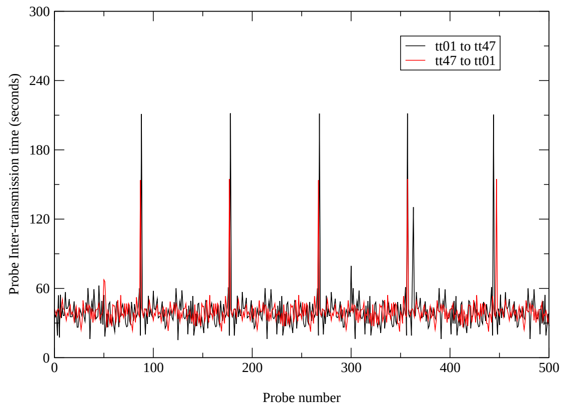

Fig. 6. TTM Inter-probe Spacing

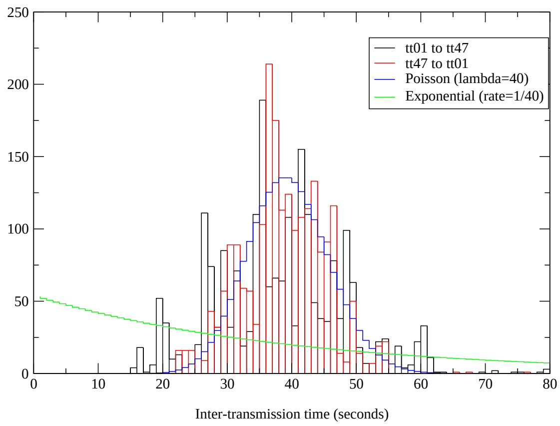

Fig. 7. TTM Inter-probe Spacing Distribution

received by three Ethernet devices. The Dag card time-stamps the packet from its hardware clock as soon as the Start of Frame Delimiter (SFD) character is detected. The packet is also received by the Ethernet NICs in both the TTM host, and the Dag measurement system. The NIC cards buffer the packet and check its CRC, possibly wait for more packets to arrive, then interrupt the host. When the host services the interrupt, it passes the received packet descriptors to the network stack, which time-stamps each packet using the system clock. Any user-space programs that are listening are then passed pointers to the packets.

On the calibration host `tcpdump` records the entire packet to disk, on the TTM host, the packet is examined. If it is a valid TTM packet, it is recorded to disk. In post processing, further tests are carried out on each recorded probe, to ensure it is suitable for analysis. If the embedded NTP quality values show the source TTM hosts NTP daemon was locked to its reference clock, then the difference between the packets received time-stamp and the source time-stamp in the packet are compared to find the One-way-Delay of that TTM packet.

For each probe arriving at the destination network, the TTM packets Id field can be found in the `tcpdump` record, and the source and destination IP addresses of that packet along with its IP Packet Id can be used to find the same packet in the Dag record. This allows us to attach the Dag time-stamp to each TTM probe Id. The data recorded by the TTM system for the experimental period was made available by RIPE NCC. These files record for each measurement probe the time-stamp of the receiving TTM host, along with the calculated TTM One-way-Delay, the TTM Id, and other information such as clock stability. By matching the probe records using the TTM Id fields between these two data sources, the arrival time-stamps for the probe packet as recorded by the Dag and TTM hosts can be compared. The difference between these time-stamps is referred to below as the *reception latency*.

A time-series of the reception latency is presented in figure 8. Since some packets will be lost in transmission, we expect to receive fewer packets on average than we receive. Host tt01 received 51,808 probes, and tt47 received 54,616.

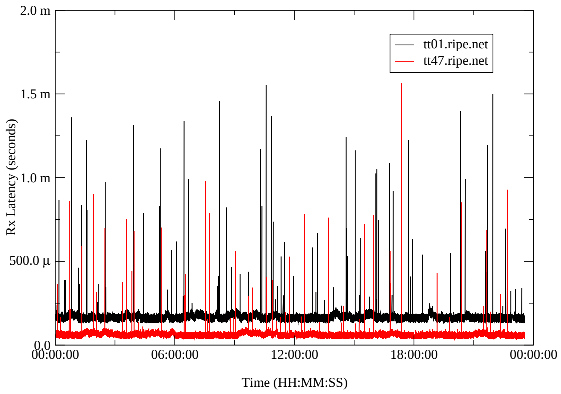

Fig. 8. TTM Reception Latency

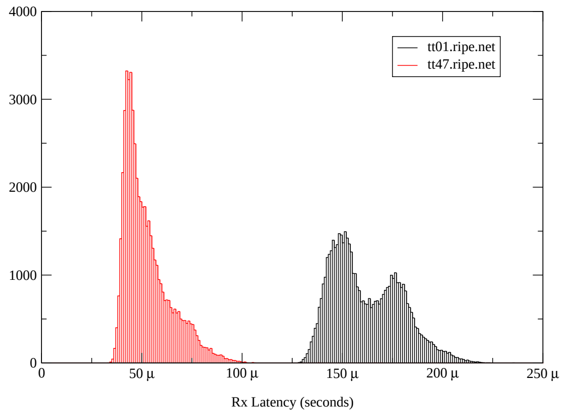

Fig. 9. TTM Reception Latency distribution

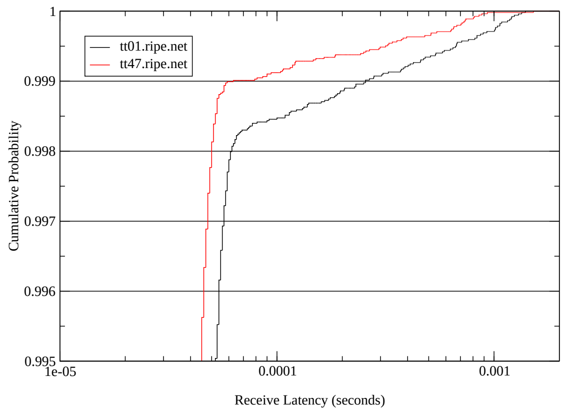

Fig. 10. TTM Reception Latency cumulative distribution

As in the previous section, the time-series shows the bulk of the reception latencies tightly constrained within a narrow band about 50μs wide, with occasional peaks to many times the median value.

A histogram of the receive latencies for tt01 and tt47 under 250μs is presented in Figure 9. Host tt01 has a median receive latency of of 158μs, and tt47 has a median of 48μs. The relative performance difference between the two hosts is even more pronounced than in Figure 4. This may be due to a difference in interrupt latency attributable to interrupt handling architecture changes between the Pentium and Pentium II CPUs. The distribution of latencies for tt01 appears to be bimodal. This could be caused by two common code paths of slightly differing lengths, or by frequent interrupt masking of a period equal to the distance between the peaks, approximately 50μs.

A cumulative histogram of the receive latency focusing on the extreme values is presented in Figure 10. In this figure, the latencies of the two hosts have been normalised to each other by subtracting from each the median latency. This is justified if we take the measured latency to consist of some fixed offset plus some varying component, and we wish to compare the varying component. Host tt01 has a latency below 1ms in 99.97% of measurements, and below 250μs in 99.89%. Host tt47 does better with only one point above 1ms, or 99.998% of points below 1ms. 99.94% of measured latencies fall below 250μs for this host. The maximum latencies were 1.55ms and 1.57ms for hosts tt01 and tt47 respectively.

The maximum receive latencies are less than the maximum transmit latencies seen in the previous section, and this likely corresponds to the impact of the scheduler. While the transmit latency includes the possibility that the scheduler will run some other process before transmitting the packet, when a packet is received the interrupt is dealt with immediately, interrupting whatever process is currently scheduled. The major components of the receive latency then must be the interrupt latency of the host, and the time taken by the host to process the interrupt. This may include copying the packet from the NIC to the hosts memory, but in modern systems the NIC often performs this operation by DMA before interrupting the host to indicate packet arrival. Other processing time may include packet integrity checks such as IP header checksum calculation.

D. End to End Comparison

Since TTM probes can be identified at both the sending and receiving end networks, it is possible to match up the transmit and receive latencies per packet, to find the total difference between the One-way-Delay as measured by the TTM system, and the wire-time One-way-Delay as measured by the Dag hardware. Table II shows the number of packets observed being exchanged between the pair of

TTM hosts tt01 and tt47 during the 24 hour experiment. Figure 11 shows the time-series of Type-P-One-Way-Delays for the experiment as recorded by the Dag cards. This a useful reference when considering the significance of the measurement errors.

TABLE II
TTM End to End Probe Experiment

From	*To*	*Sent*	*Received*	*Loss Rate*
tt01	tt47	2160	2121	1.81%
tt47	tt01	2160	2154	0.28%

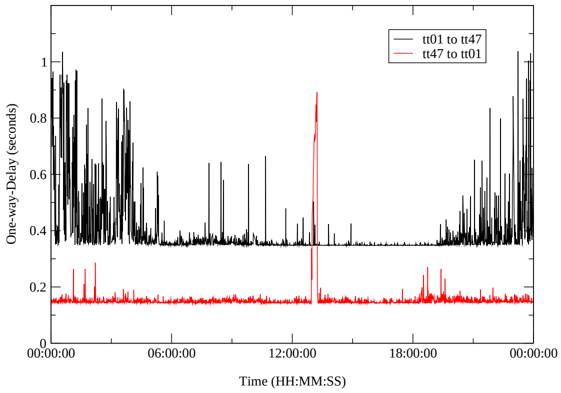

Fig. 11. TTM Type-P-One-way-Delay

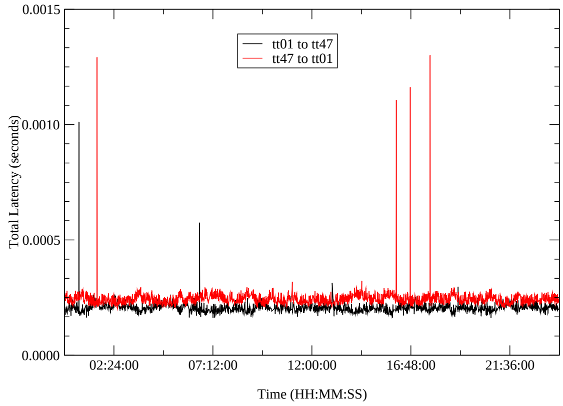

Fig. 12. TTM Total Latency

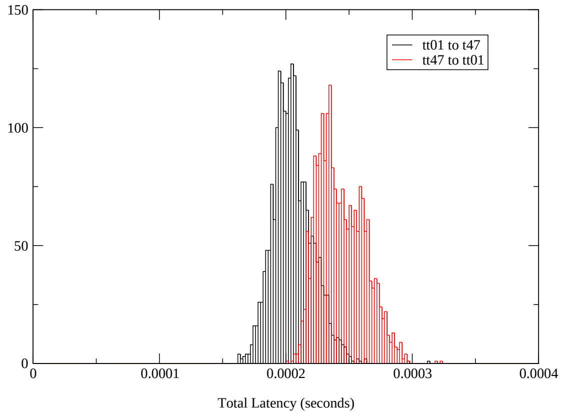

Fig. 13. TTM Total Latency distribution

Figure 12 shows the time-series of the sums of the transmission and reception latencies per packet for all the TTM probes that were successfully recorded at both sites. Figure 13 is a histogram from 0 to 400μs in 2μs bins of the time-series data. The median total latency in the tt01 to tt47 measurement is 204μs, while in the tt47 to tt01 direction the median is 240μs. The distribution about these medians in both cases is of similar shape, and falls almost entirely within plus or minus 50μs. There are some excursions to extreme values, with total latency reaching a maximum of 1.3ms. This is to be expected from the previous examination of the bulk properties of all of the probes transmitted and received by each host, where we attribute these to host behaviour. The bimodal nature of the tt47 to tt01 latency is due to the receive behaviour of tt01, as seen in figure 8.

It is immediately noticeable that the distributions of total latency are not equivalent, despite being captured by the same pair of hosts. This is due to the fact that while one host is generally faster then the other, the relative speedup in the transmission and reception processes between the machine is not even. Since the improvement in reception latency is greater than the speedup in transmission, the measurement in which the faster host (tt47) is receiving, (ie. tt01 to tt47) has the lower total latency.

The total time added to the TTM One-way-Delay measurements by the host behaviour and the variation in this added time is as expected independent of the magnitude of the One-way-Delay. The total latency experienced is dependent on the particular hosts used, and will be largely independent of network conditions, although cross traffic or collisions at the TTM hosts Ethernet ports can cause additional latency up to at least one maximum length Ethernet frame per host.

The extreme values are always positive, that is in the measured end to end delays there appear to be no delays measured by the TTM system that are less then the wire-time One-way-Delay measured by the Dag system. This indicates that the TTM measurement of the minimum delay over a path is likely to be reliable, although it will only be accurate to within the 50μs variation in TTM accuracy observed.

Extreme values are rare; there are only two probes for which the total latency is more than 100μs from its mean for the the tt01 to tt47 measurement, and four in the opposite direction. For this experiment, an accuracy of better than plus or minus 100μs can be claimed for 99.9% and 99.8% of probes respectively. Due to the small number of exceptional values observed, these numbers are not significant, but reference to the previous sections investigating transmission and reception latency individually tend to support them.

A term e for random error is defined for Type-P-One-way-Delay[3]. The difference between the wire One-way-Delay and the recorded One-way-Delay should be measured for the system under test repeatedly. The 95% confidence interval would then be the range from the 2.5th to the

97.5th percentile of the deviations about the mean, which is the systematic error. The calibration error e could then be the largest absolute value of these percentiles, plus the clock related uncertainty. The value of e for both pairs of source and destination in this experiment is derived in table III.

TABLE III
TTM error distribution

From	*To*	*2.5th* (μs)	*Median* (μs)	*97.5th* (μs)	*e* (μs)
tt01	tt47	177	204	239	35
tt47	tt01	215	240	282	42

V. Conclusions

Passive measurement systems with inherently higher accuracy than commodity PC NIC based solutions can be used to measure and characterise per host and end to end errors in deployed active delay measurement systems, even in cases where the deployment is inter-continental. This is an important verification exercise, and provides calibration of operational systems that is not otherwise possible.

In the case of RIPE NCCs TTM system, the measured errors showed some expected systematic offset, and a variable error of less than 50μs at a 95% confidence level. These measurements of error in the deployed operational system agree well with laboratory bench tests of directly connected TTM hosts. This finding validates the use of back-to-back testing of delay systems, provided that care is taken to replicate the operational loading conditions on the tested systems.

When hardware based passive measurement equipment is used in conjunction with such back to back testing on the laboratory bench, further insight into the overall latency behaviour becomes possible. By providing information on the transmission and reception latency separately for instance, it becomes possible to explain the asymmetry in One-way-Delays between two hosts under test. Furthermore, given the good agreement between laboratory and deployed tests, the possibility of performing calibration measurements of the distributions of transmission and reception latency on individual test-boxes before they are deployed is raised.

The extra costs involved in hardware based passive measurement systems may not be justified for operational use when building a large geographic scale delay measurement system, as the active software approach provides a high level of accuracy when compared to the magnitude of the delays measured. The precision and accuracy of passive hardware systems however is indispensable when studying network events at shorter time-scales, such as the queueing behaviour of routers. When studying events at such fine time-scales, the union of active and passive techniques provides a powerful tool.

References

[1] David L. Mills, "Network time protocol (version 3)," Tech. Rep. RFC1305, IETF, March 1992.

[2] Trimble Navigation Limited Timing and Synchronization Group, *Palisade NTP Synchronization Kit User Guide*, June 1999, Part Number 39139-00.

[3] G. Almes, S. Kalidindi, and M. Zekauskas, "A one-way delay metric for ippm," Tech. Rep. RFC2679, IETF, September 1999.

[4] Sunil Kalidindi and Matthew J. Zekauskas, "Surveyor: An infrastructure for internet performance measurements," INET'99 Online Proceedings, June 1999, http://www.isoc.org/inet99/proceedings/4h/4h_2.htm.

[5] Henk Uijterwaal and Olaf Kolkman, "Internet delay measurements using test traffic," Tech. Rep. RIPE-158, RIPE NCC, May 1997.

[6] I. D. Graham, S. F. Donnelly, S. Martin, J. Martens, and J. G. Cleary, "Nonintrusive and accurate measurement of unidirectional delay and delay variation on the internet," INET'98 Online Proceedings, July 1998, http://www.isoc.org/inet98/proceedings/6g/6g_2.htm.

[7] Philip Tree, *Dag 3 Installation Guide*, The University of Waikato, August 2000, http://dag.cs.waikato.ac.nz/dag/docs/dig_v2.1.pdf.

[8] DEC, Intel, and Xerox, "Ethernet local area network specification version 2.0," November 1982, DEC Part Number: AA-K759B-TK.

[9] V. Paxson, G. Almes, J. Mahdavi, and M. Mathis, "Framework for ip performance metrics," Tech. Rep. RFC2330, IETF, May 1998.