Accuracy and Expressiveness in Adaptive Bandwidth Measurements

Marc Pucci
Telcordia Technologies, Inc.
(Joint work with James L. Alberi, Allen McIntosh and Thomas Raleigh)

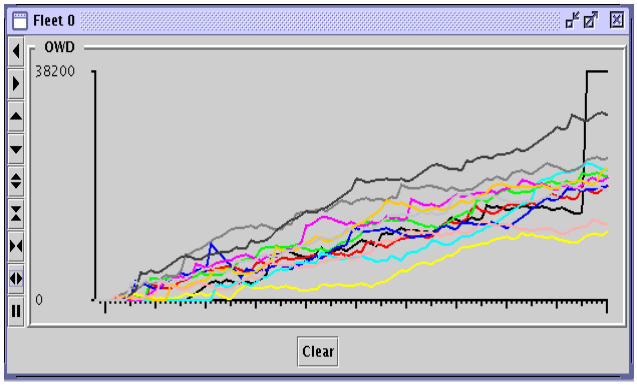
Adaptive bandwidth techniques severely stress the capabilities of active Internet measurement platforms. Beyond requiring precisely timed ingress and egress packets, they also demand accurate spacing between packets as they are emitted. This level of control is unattainable using purely program-driven mechanisms such as busy-waits in user code. This becomes especially evident with very fast link speeds where measurement variance can obscure measurement data. Furthermore, no single packet generation profile for adaptive measurements has been agreed upon by the community, nor is it likely that a single specification will be appropriate for all classes of cross traffic, link speeds or application sensitivity. Thus we see a need for a highly expressive and tailorable description of probing sequences, and a sufficiently efficient generation mechanism, to improve the utility of these measurements.

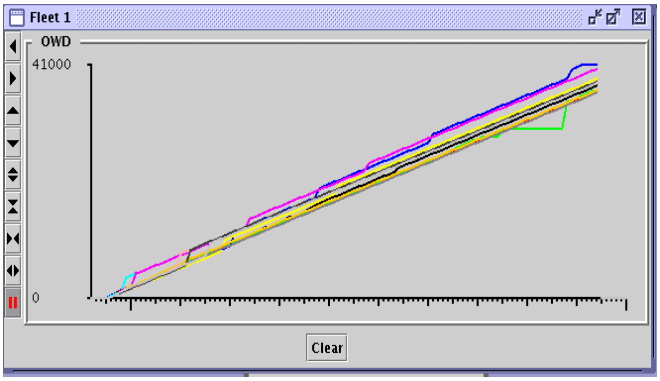
Accuracy

To support our interest in comparing different forms of adaptive bandwidth techniques, we continue to explore these two areas in great detail. While the commodity nature of GPS and CDMA based timing sources has improved the accuracy of active one-way measurements, it does not improve the generation of the controlled sequences of packets that are fundamental to active bandwidth estimation techniques. Since the accuracy of such measurements depends on the egress packets having precise, repeatable timing gaps between packets in a probe, the lack of such a facility will always compromise the overall reliability of the measurement.

It is possible to achieve highly accurate packet spacing with custom hardware, though we feel that this would substantially limit the availability and accessibility of measurement nodes. We believe that judicious use of existing network interface cards and enhancements to basic operating system functionality will yield precision consistent with the needs of adaptive measurement processes.

The figure below illustrates the improvement possible when moving from a user level programmed busy-wait to a mechanism based on using internal PC system timer hardware and Linux operating system enhancements. The plots show the send and receive times of 12 packet trains and illustrate the improvement in measurement variability between the busy-wait loop on the left and the enhanced OS-based version on the right. By moving time-sensitive functions into the kernel, disruptions due to events beyond the direct control of the user, such as operating system context switches and device interrupts, are alleviated. Without this reduction in variance, accuracy of the process becomes suspect.





This technique of using kernel enhancements is sufficient for moderate line speeds up to 100 megabits, where timing accuracy on the order of 122 microseconds is acceptable. In the gigabit range, kernel interrupts are inadequate in retaining the necessary degree of accuracy. In this domain we rely upon the ability of some network cards to sequence packet emissions directly. These cards have hardware FIFOs that can be synchronized against the internal high-speed clock of the card to obtain inter-packet gaps on the order of 10's of nanoseconds. With interval control of this magnitude, it is possible to generate timing patterns that are detailed enough to estimate available bandwidth in very high speed networks without resorting to custom hardware. We are currently evaluating the efficacy of this technique.

Expressiveness and Ease of Control

Bandwidth estimation techniques expose a need for greater expressiveness in the specification of the packets that make up a measurement probe pattern. We have witnessed a variety of patterns used in the estimators developed thus far. We propose the use of a simple descriptive representation of a desired pattern to guide the packet generation process. Additionally, by adding a means for interacting with the requesting (measuring) side process, we can improve the overall measurement utility. This structure also permits a more accessible comparison of different techniques, since we can readily swap in different algorithms while leaving the measurement infrastructure intact.

Bandwidth estimators are, by their very nature, extremely dynamic in their execution. They need to adapt to intermediate measurements as they zero-in on the best possible estimate of traffic conditions. We suspect that no single pattern will be sufficient for all possible combinations of line speeds, cross traffic and perhaps even for the type of application that an estimation is relevant to.

The use of a language to describe the probe profile also implies that we need not create multiple specialized packet generators. Instead, a single, generalized packet emitter can be programmed and tuned in real-time to adapt to changing requirements of a measurement algorithm. We do not need to alter the code running in remote measurement platforms to add a new emitter for a new algorithm. The emitter accepts a packet specification and customizes itself to the described sequence as required by the measurement algorithm.

The language we have explored contains basic elements for describing the size, content and timing relationships between packets, all of which can be varied in real-time using a synchronization mechanism that ties the generator back to the measurement processor. The instruction set is intentionally limited so that run-time execution requirements are minimal and can be kernel resident. Efficiency is maintained by encouraging the interaction between the generator and the measurement process to be a co-routine relationship. Only enough generator activity occurs to create descriptors for one burst of packets, which are then passed to the packet sequencer described above. We are using this mechanism as the basis of a measurement infrastructure for a variety of active measurement techniques.