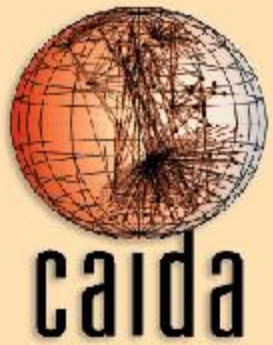


bandwidth estimation:
measurement methodologies and applications

february 2003

kc claffy, ucsd/sdsc/caida
kc@caida.org

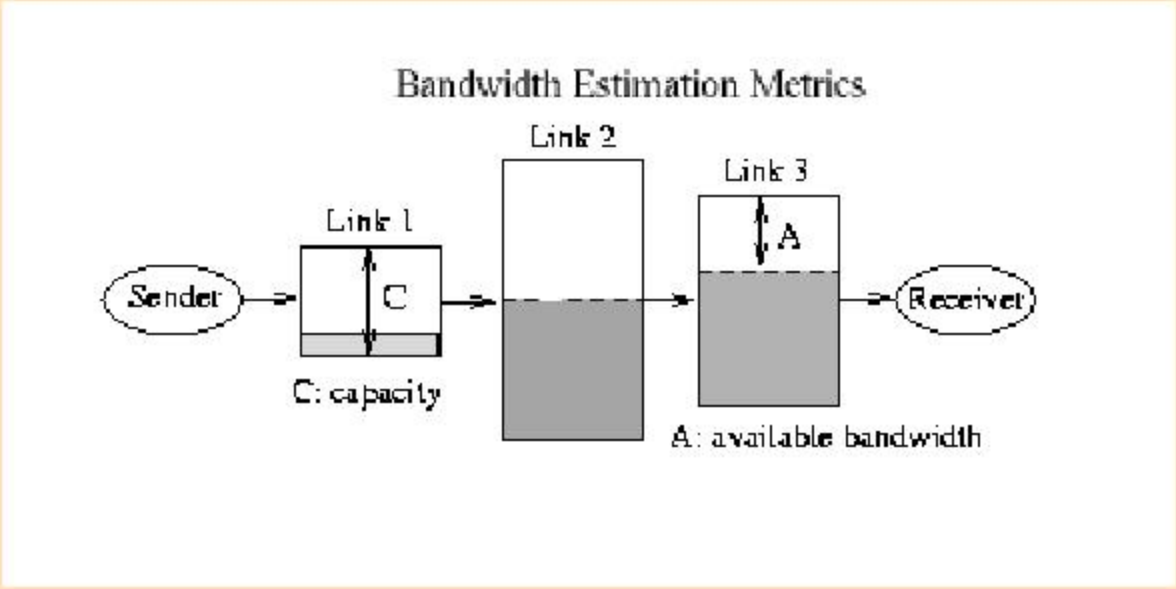
constantinos dovrolis, georgia institute of technology
dovrolis@cc.gatech.edu

motivation

who needs better bandwidth estimation?

- high throughput data-intensive applications
- scientific visualization
- colaboratory interactions
- remote sensor analysis

background: metrics definitions



- capacity
  - narrow link
- available bandwidth
  - tight link
- TCP throughput

background: existing tools

- capacity (end-to-end)
  - pipechar (?)
  - sprobe
  - pathrate
- capacity (per hop)
  - pathchar
  - pchar
  - clink
- available bandwidth
  - pathload
  - pipechar (?)
- TCP throughput
  - netest-2 (?)
  - treno
  - iperf

goal: calibrate existing tools

- how do tools perform against cross-traffic?
- what are requirements for running tools?
  - logistical
  - security
- what is the overhead?
  - time-to-measure
  - injected traffic
- are the results valid?

testing methodology

- install tools on two freebsd end hosts
- use calngi lab
  - fully controlled environment
  - varying test topology
    - **100M links**
    - **GigEther links**
  - baseline measurements with no cross-traffic
  - simulating cross-traffic as pseudo-random traffic flows
    - **SmartBits hardware**
    - **SmartFlow software**

lab testing: simulation of cross-traffic

simulated cross-traffic by SmartBits/SmartFlow

- basic parameters:
  - number of flows, N
  - packet size for each flow (constant), p
  - order of flows round-robin to approximate pseudo-random traffic
  - percentage of the link utilization, 10% < u < 90%
  - burst size, b <= N
- application
  - sends b flows back-to-back (burst)
    - takes time t
  - waits for time d = t*(100-u)/u
  - repeats the burst, cycling the pre-determined flows
- link transmits bits for u% of the time on average

lab testing results

- layer 2 store-and-forward devices affect tools accuracy
  - not handled by pathchar, pchar, clink

C. Dovrolis, R.S. Prasad and B. Mah,
"the effect of layer-2 store-and-forward devices on per-hop capacity estimation"(ieee infocom 2003)

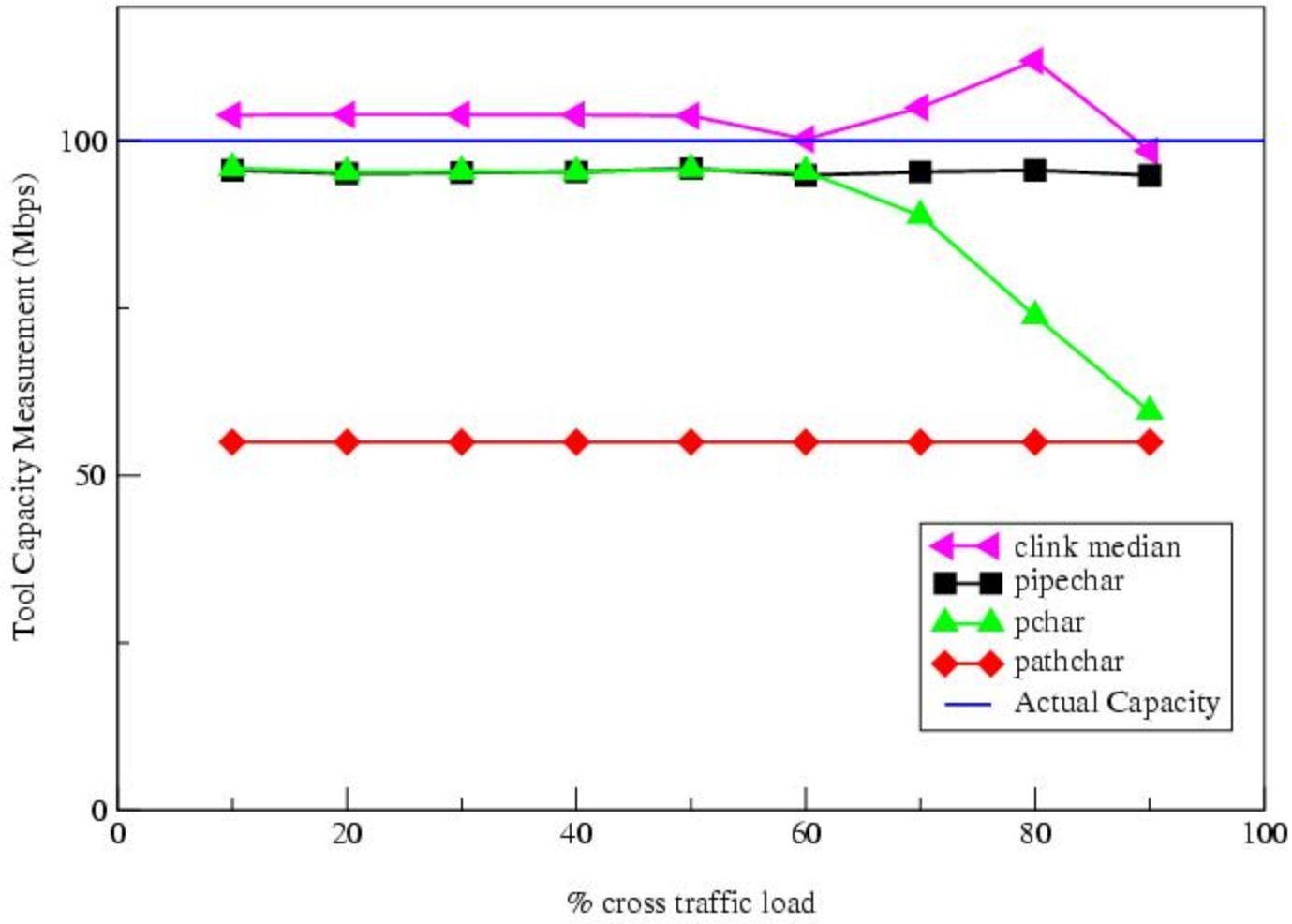
- not all routers are created equal
  - internal switches
  - different buffer configurations
  - use of "slow path" for ICMP traffic

- accuracy of tools deteriorates at high loads
  - we have shown that per-hop capacity estimation tools (pathchar, pchar, clink) produce consistent and significant errors when path includes layer-2 store-and-forward devices.
  - all paths we have experimented with include one or more such devices
  - real and important problem
  - paper available on const web page and at infocom 2003

upshot: testing bandwidth estimation tools is nontrivial

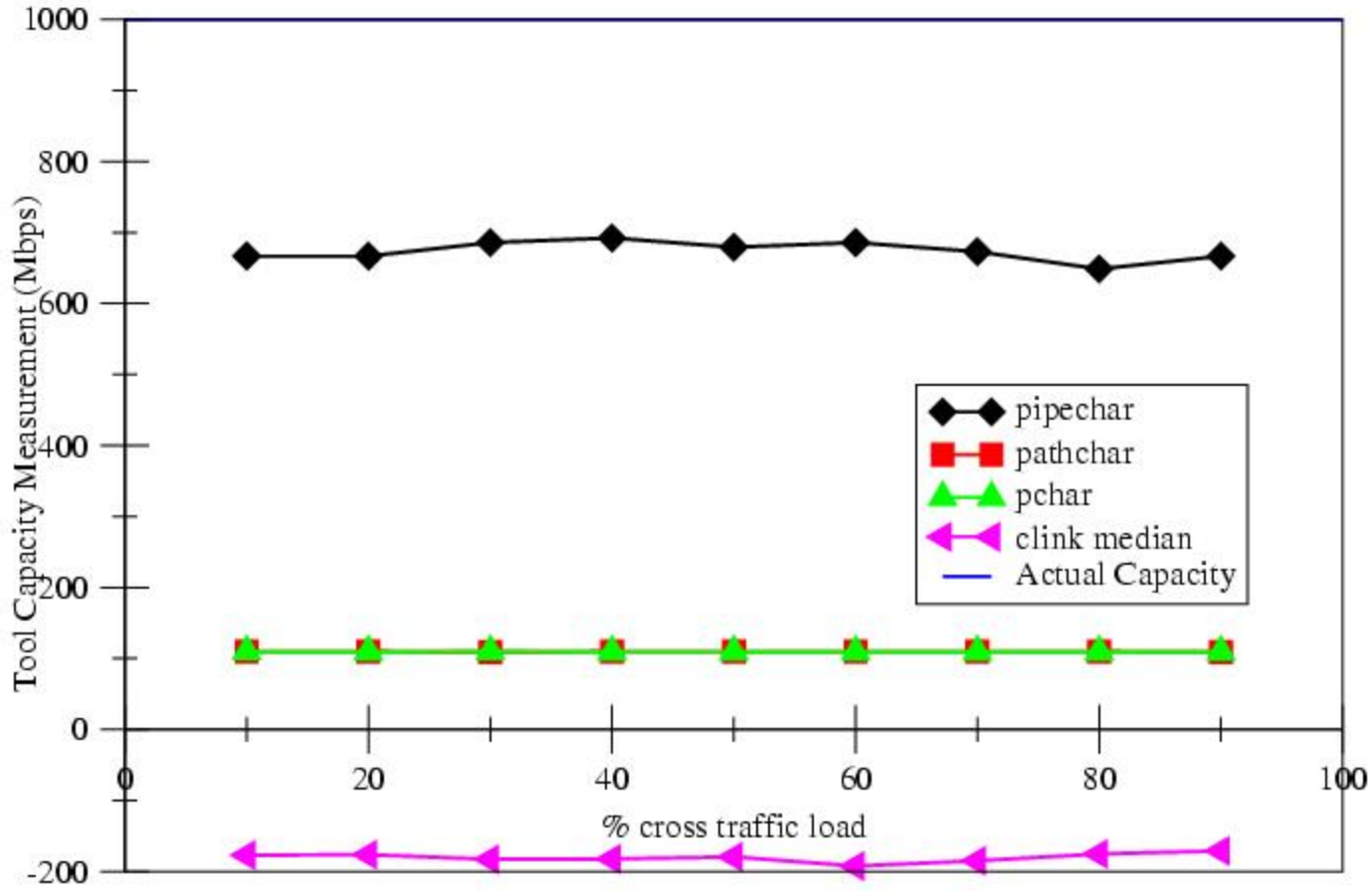
lab testing: layer 2 devices foil bwest tools

100Mbps

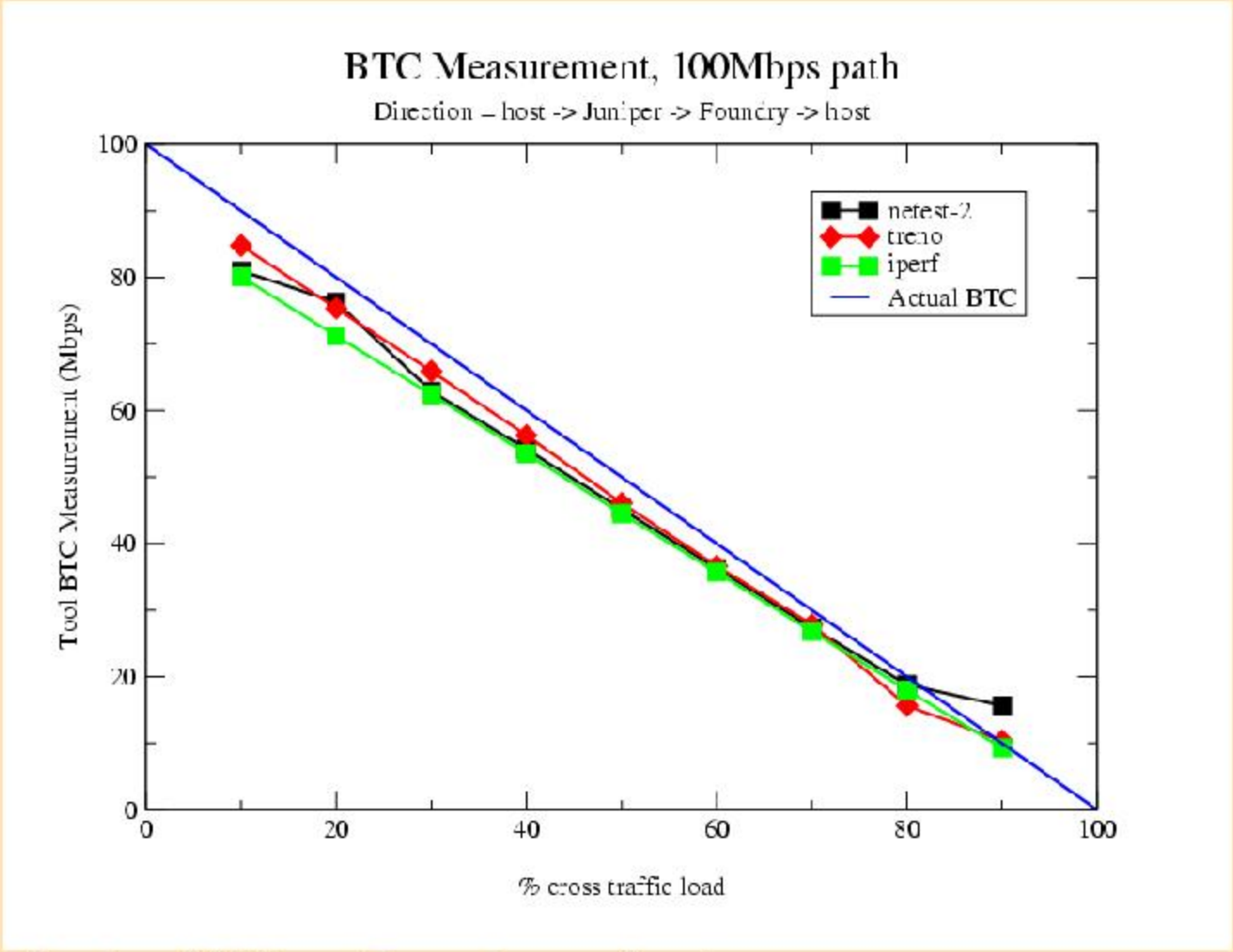


lab testing: layer 2 devices foil bwest tools

gigE

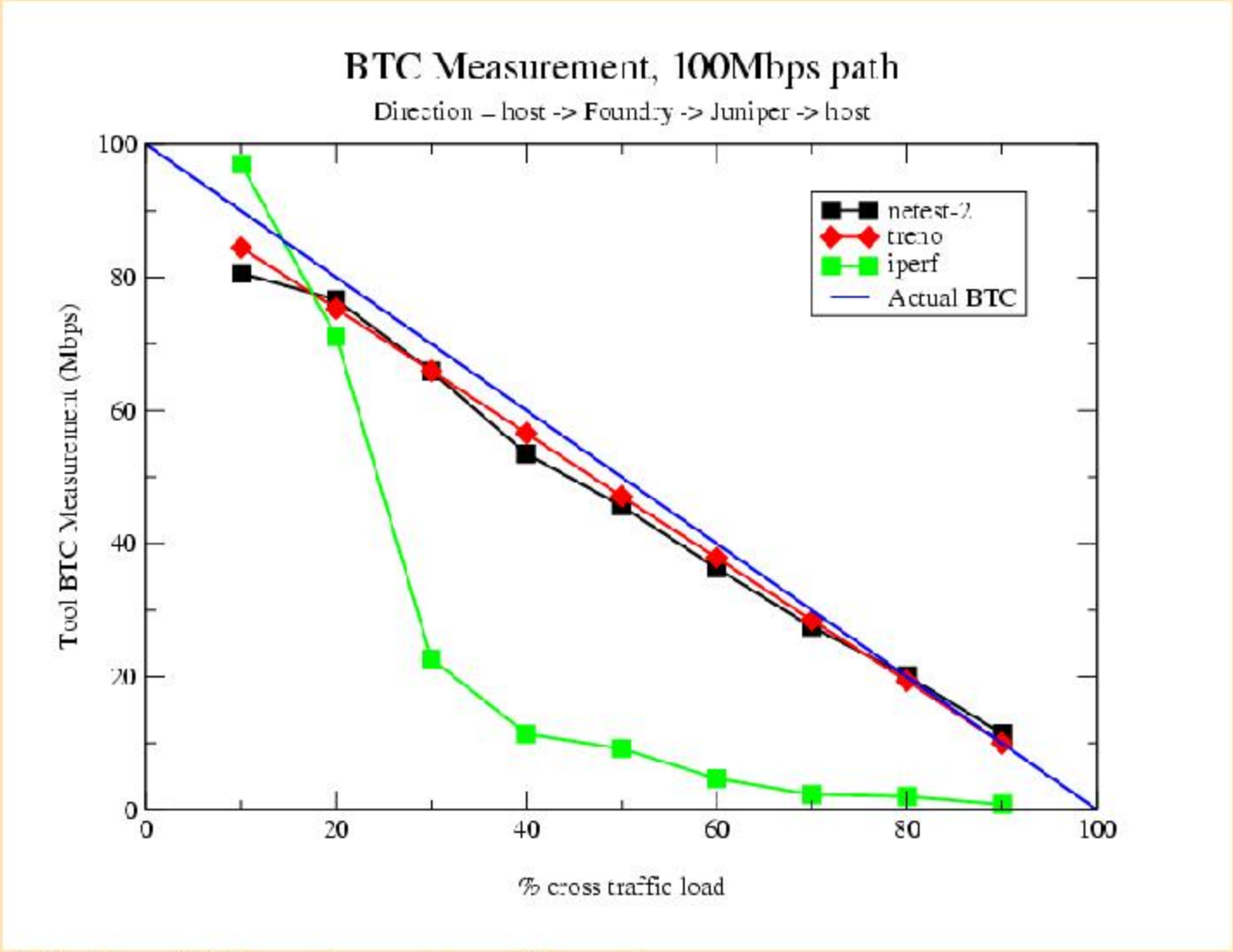


lab testing: router configuration matters



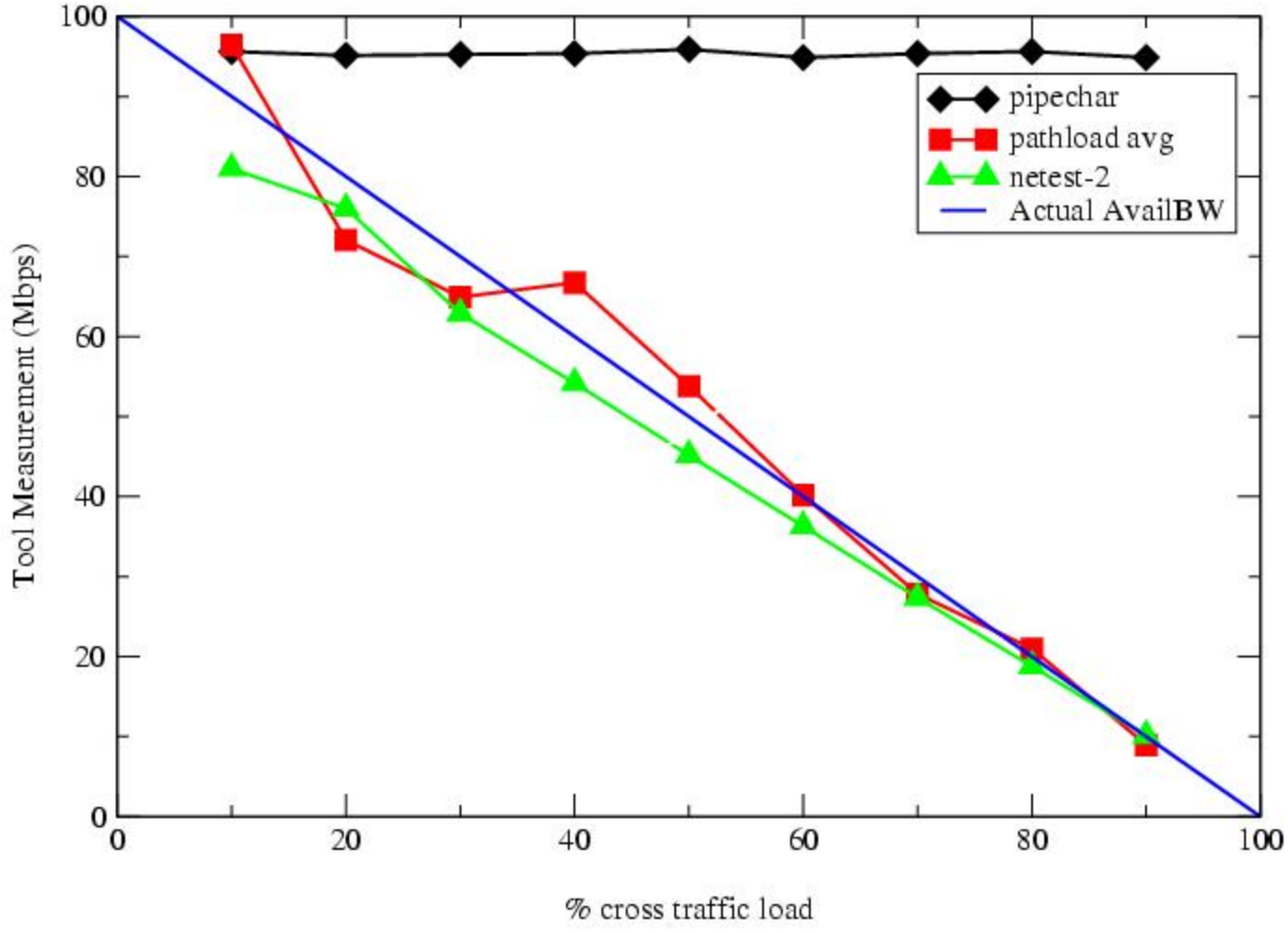
- host -> JuniperM20 -> Foundry -> host

lab testing: router configuration matters

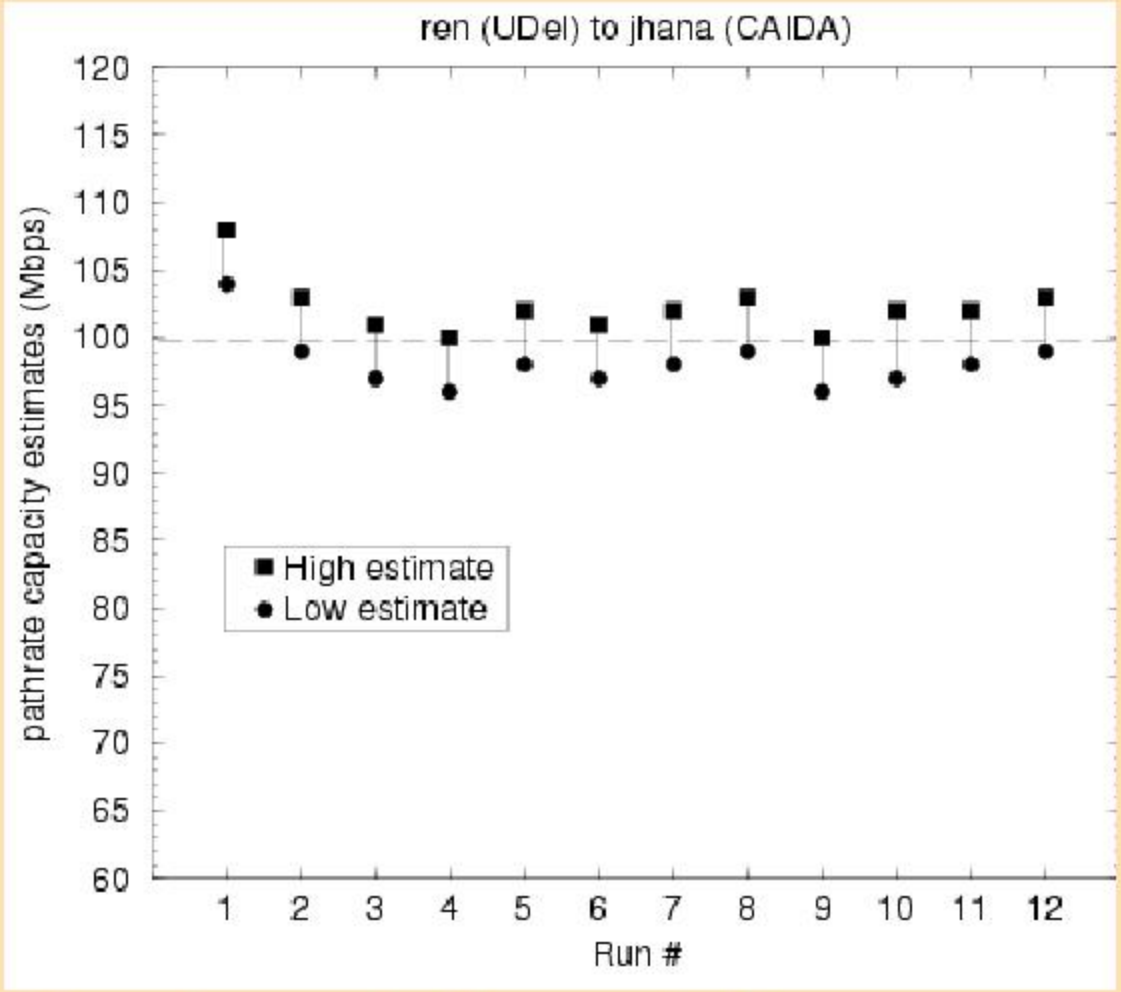


- identical links, different order of routers
- host -> Foundry -> JuniperM20 -> host

lab testing: cross-traffic sensitivity

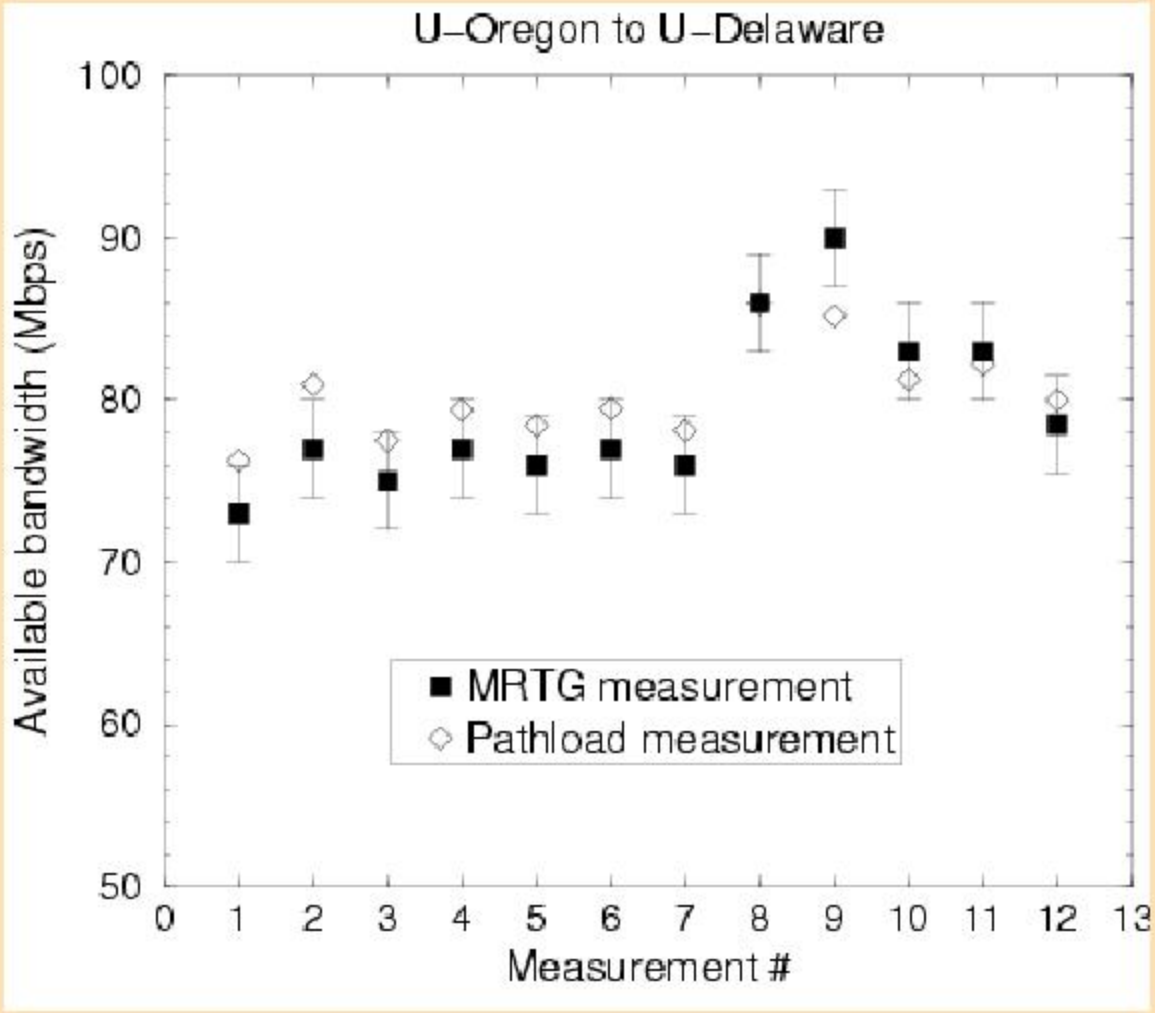


gatech tools: pathrate



- measures end-to-end capacity (narrow link)
- more reliable, accurate than pathchar, pchar, clink

gatech tools: pathload



- measures available bandwidth (tight link), fast w minimal intrusion
- uses new self-loading periodic streams (SLoPS) algorithm
  - C. Dovrolis and M.Jain, "End-to-End Available Bandwidth: Measurement methodology, Dynamics, and Relation with TCP Throughput" SIGCOMM 2002

problems at high bandwidth

- working on getting pathrate and pathload to work on high-bandwidth paths (600Mbps to 1Gbps)
- pipechar measurements have shown clearly that pipechar does not measure available bandwidth, but rather capacity
  - problematic, since tool claims otherwise
- both pathrate and pathload have been tested successfully in paths limited by OC-12 (640Mbps) link
- gigE paths still challenging when network interfaces batch interrupts. we have ideas on how to fix the problem..
  - more about this at PI meeting in March

important for DOE PIs to
give us feedback w our tools!

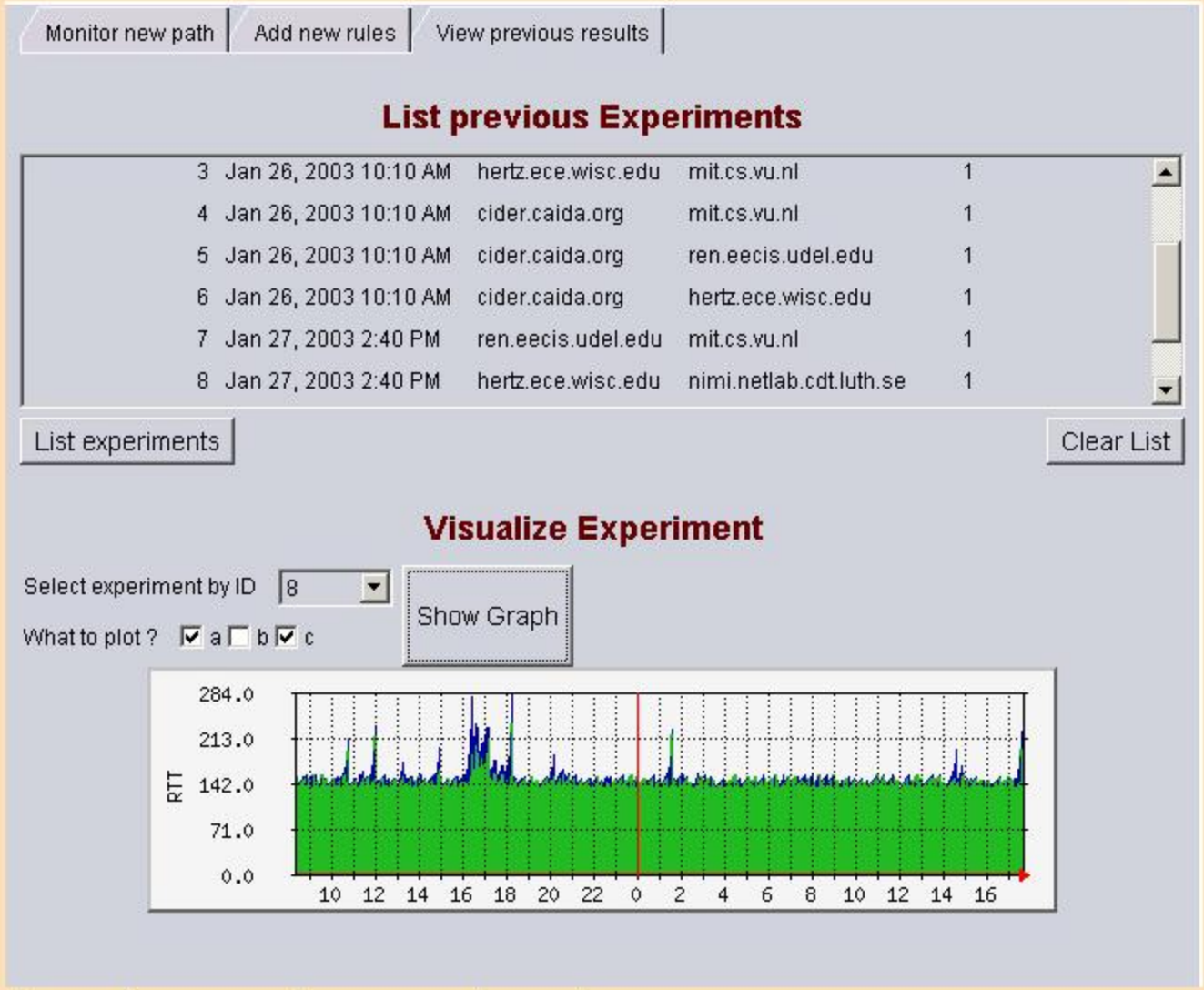
bandwidth estimation tools gui

ANEMOS architecture

- clients (web applets)
  - select a path to monitor
  - add new rules for data analysis
  - view previous results
- workers
  - execute bwest tools
- coordinator
  - manages workers
  - maintains scheduling queue for active measurements
  - analyzes measurements
  - issues user alarms
  - archives results

tools: bwest gui: even a user interface!

ANEMOS: an Autonomous NEtwork MOnitoring System



- gui for network operators and end-users

tools: bwest gui features

ANEMOS (to be presented at PAM03)

- coordinated measurements of delay/loss/avbw in multiple paths
- GUI for monitoring results through MRTG graphs
- resulting measurements archived with MySQL, retrieved with GUI-specified queries.
- user can specify "rules" on the resulting measurements for automatic processing and alarm detection (e.g., issue alarm if RTT > 100msec and avbw < 2Mbps in paths X and Y).
- code available in april 2003

need scidac researchers' help

use and give us feedback on our tools please!

- many recent scidac-related publications cite pathcar or pipechar and not this project's tools.
- work on optimizing bulk TCP transfers in high bandwidth-delay product paths:
  - Dunigan, Mathis, Feng, Tierney, Rice
  - why? they all need better bandwidth estimation
- important to integrate w scidac projects
  - especially for improving TCP or applications

future directions: research

Internet spectroscopy

- observe cell or slot-based traffic on broadband links
- passive measurement
  - does not require additional probes
- analysis of inter-packet delay distributions
  - radon transform
  - entropy minimization
- infer specific layer 2 provisioned bandwidth
  - based on subtle "noise" features

future directions: engineering

- use of bandwidth estimation in maximizing bulk TCP throughput in high bandwidth-delay product paths.
- we demonstrate that significant gains result when the socket buffer size is set based on bwest measurements.
- application-layer technique for automatic socket buffer sizing, based on bwest. technique does not require any TCP or OS changes
- annual bwest workshop planned summer 2003

acknowledgements

- PIs
  - constontinos dovrolis
  - kc claffy
- key admin and technical personnel
  - margaret murray
  - nevil brownlee
- awesome grad students
  - ravi prasad (pathrate, effects of layer-2 devices on pathchar-like tools)
  - manish jain (pathload)
  - antonios danalis (anemos)

http://www.caida.org/projects/bwest/

margaret murray, marg@caida.org