On Dynamics of Network Transport Protocols

Nageswara S. Rao

Computer Science and Mathematics Division

Oak Ridge National Laboratory

Leon O. Chua

Department of Electrical Engineering and Computer Science

University of California at Berkeley

Workshop on Signal Processing, Communications, Chaos and Systems

Newport, RI, June 20, 2002

Sponsored by

DARPA Network Modeling and Simulation Program

DOE MICS High-Performance Networking program

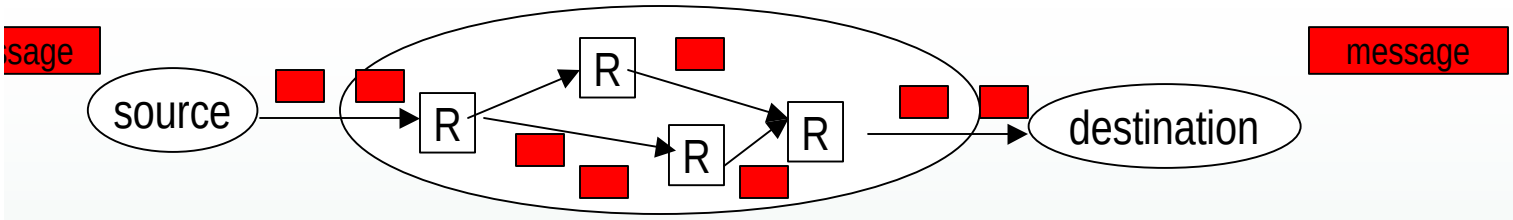
UT-BATTELLE

line of Presentation

- TCP Simplified Dynamics
- Composition Model
- Experimental Results

work Transport (Simplified View)

- Messages from source get broken into packets



- Packets are routed over the network
  - Routes are decided by routers and service providers in Internet
  - Source has very little control on paths traversed by packets
- Most network transport is controlled by two protocols
  - TCP: Transport Control Protocol
    - Ensures reliable delivery – retransmits delayed or lost packets
    - Vast majority of Internet traffic
  - UDP: User Datagram Protocol
    - Sender puts datagram on the network to be routed
    - No further action is taken –lost packets are lost forever

End-to-End Transport Dynamics are Important

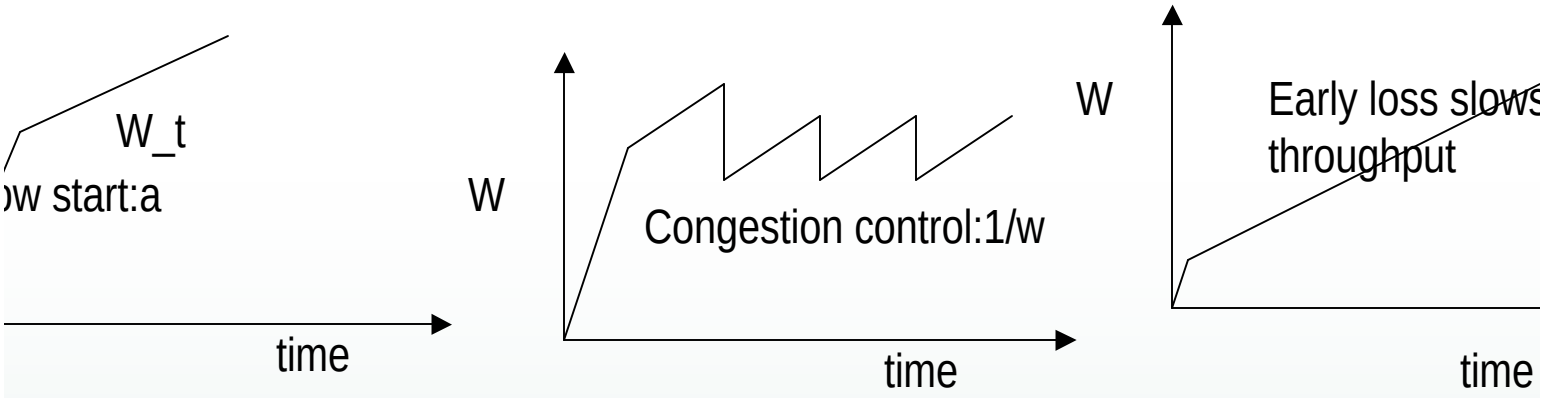
- TCP performance: High bandwidth for large data transfers
  - initial transients can force TCP into congestion control, achievable bandwi is very limited
- Control of end devices: Remote control of instruments
  - Jittery dynamics will destabilize the control loops
  - Will not be able to effectively execute interactive simulations

Very Limited insights about end-to-end dynamics

- Chaotic TCP window dynamics (Veres)

But observations are a give-away

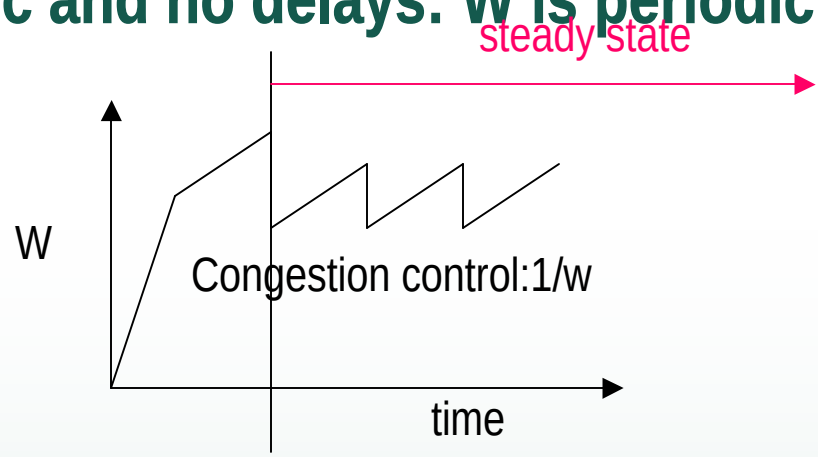
ditional View: Dynamics of TCP



ransport Control Protocol (TCP) Outline

- Uses window mechanism to send W bytes/sec
- Dynamically adjusts W to network and receiver state
  - Keeps increasing is no loses
  - Keeps shrinking if losses are detected
- Slow start phase:
  - W increase exponentially until W_t or loss
- Congestion Control:
  - linearly increase W with delivered packets
  - Multiplicatively decrease with loss

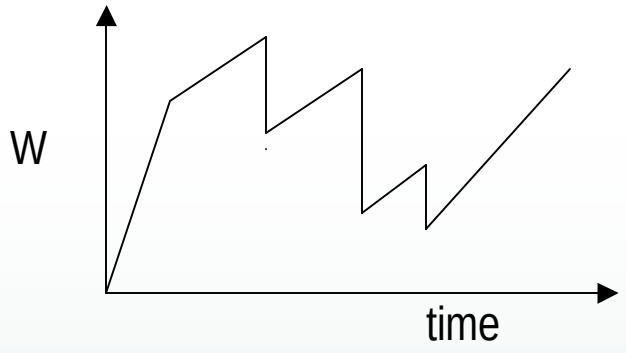
Competing Traffic and no delays: W is periodic



CP has a stable cycle under no network delays

- Prolonged Unstable regime:
  - W bytes increases with acknowledgments
- Instantaneous stabilization:
  - Bring W to a lower value with a loss as soon as bottleneck BW is exceeded

npeting traffic delays: W is not guaranteed to be periodic



vork delays:

otification of losses are delayed

Meanwhile packets are sent which will overflow – multiple losses

- W bytes increases with acknowledgments

all buffers at routers:

ackets exceeding the bandwidth are buffered

Network delays and competing traffic will generate complicated profiles of W

npeting traffic, Network Delay, Router Buffers

omplicated traffic can generate complicated TCP dynamics

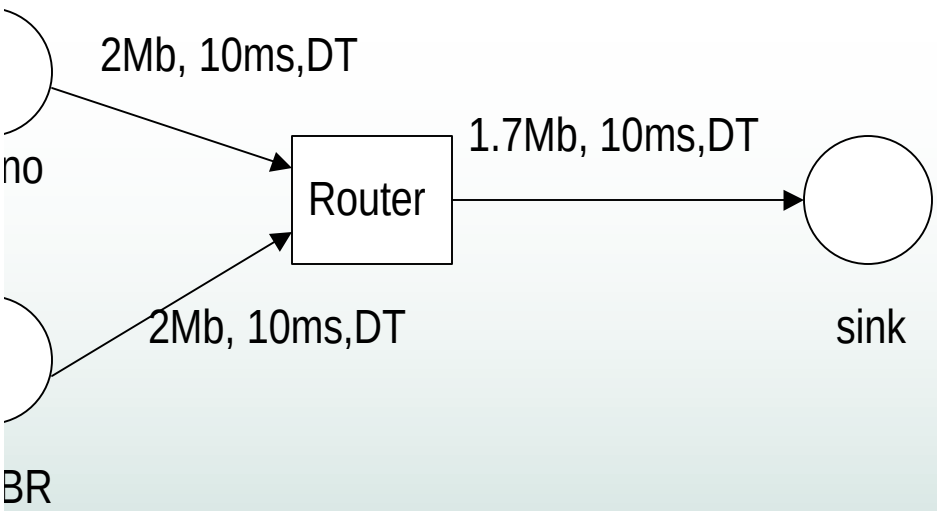
uestion: Can simple traffic generate complicated trajectories ?

Yes, under certain conditions

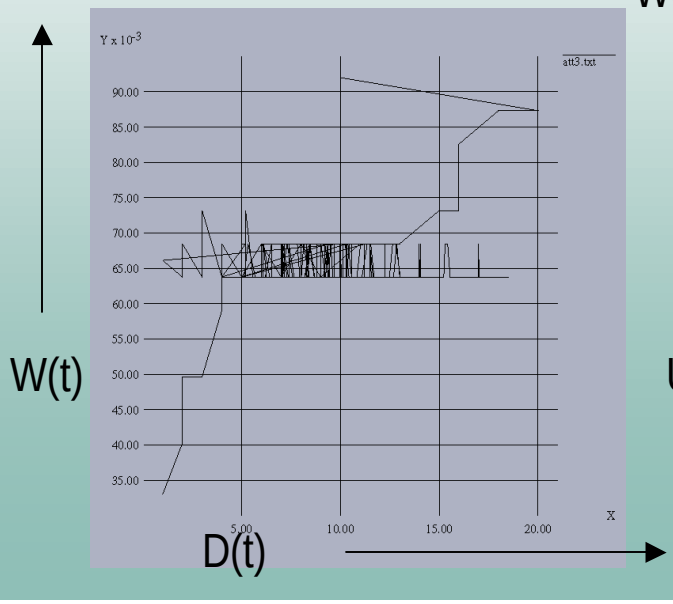
- Net-Effect: The stabilization part of the ideal case is prolonged, and is replaced by another unstable regime.
- To see this effect we introduce end-to-end packet delay

tate-Space of TCP:

P Competing with UDP
simulation)

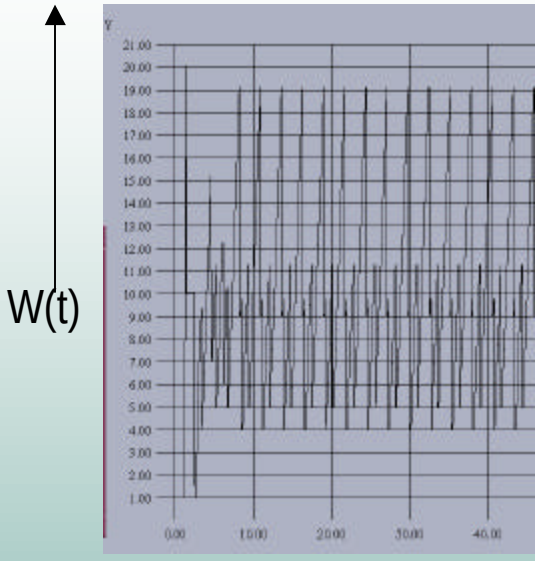


are phase plot:
w-size W(t) vs.
d-to-end delay D(t)



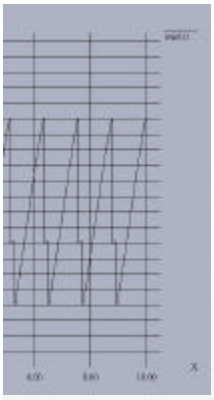
As CBR rate is varied

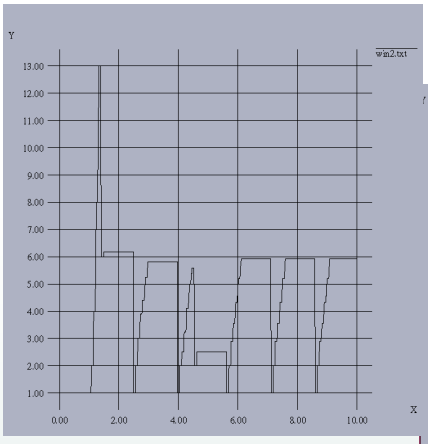
TCP competing with UDP/C
the router generates a v
of dynamics

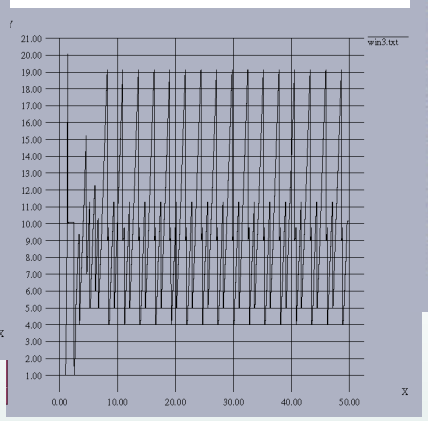


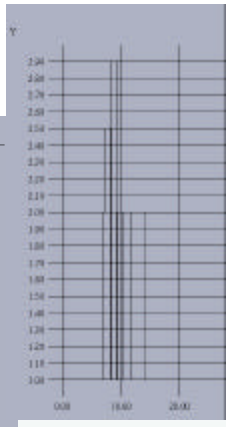
UDP/CBR=1Mbs

P Competing with UDP





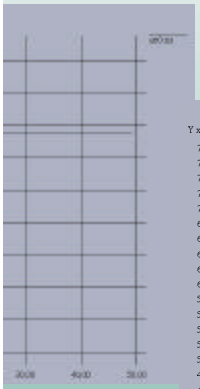




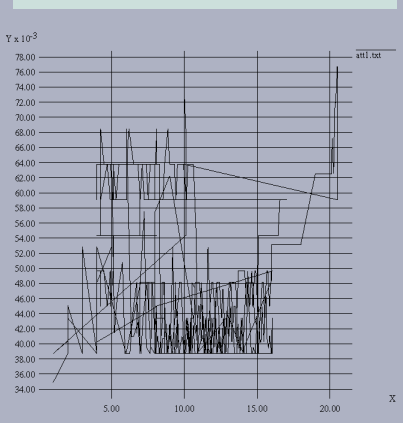
R: 0Mbs

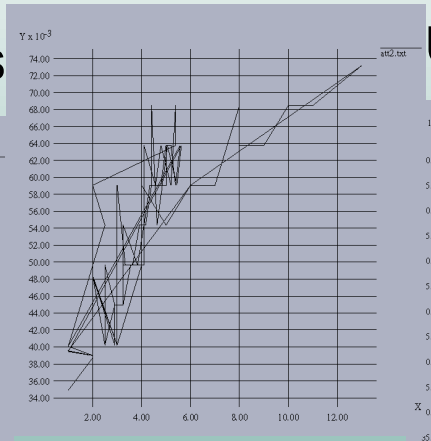
UDP/CBR: 1.0Mbs

UDP/CBR:

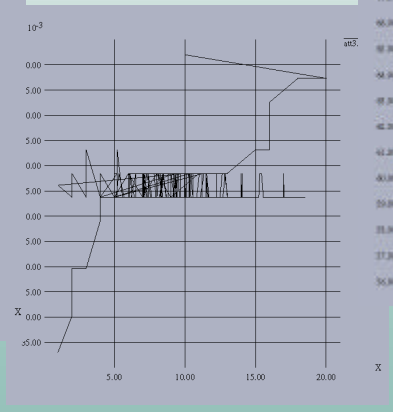


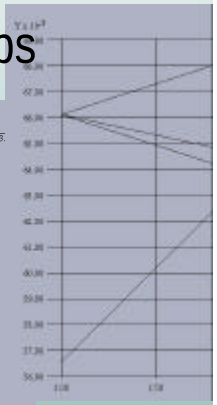
UDP/CBR: 0.5Mbs





UDP/CBR: 1.7Mbs

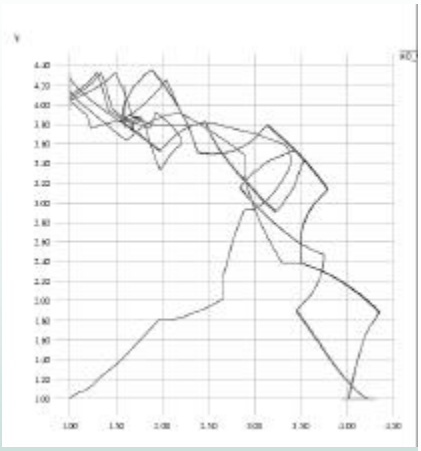




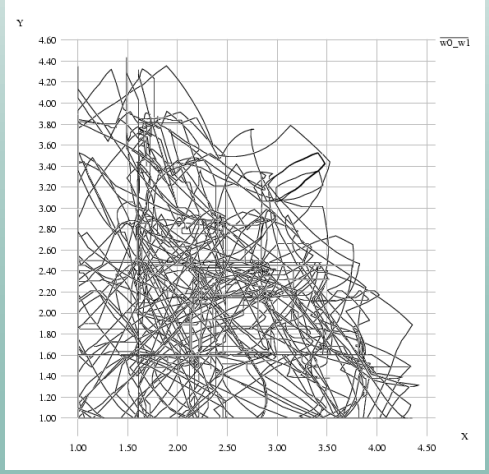
Chaotic Dynamics of TCP

Competing TCP streams: Window dynamics are chaotic

- Hard to predict – resemblance to random noise
- Hard to conclude from experiments – nearby orbits move faraway later
- Hard to characterize – chaotic attractor
  - Poincare map of two window sizes
    - Two-streams case
    - Four streams case



eres (2000) did not rigorously
stablish chaos in a formal sense
Attractor could have been
generated by periodic orbit with large period
/e are attempting to show
napback repellers in TCP equations
formal result

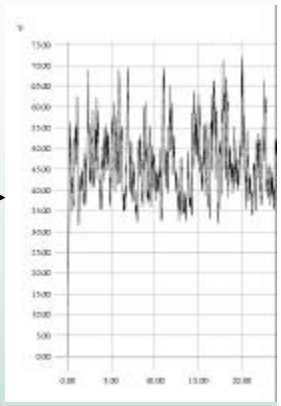


Noisy Nature of TCP
(simulation)

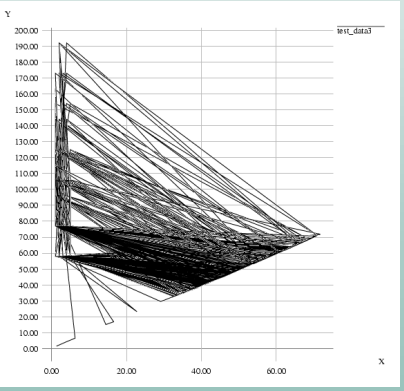
TCP source → Router: uniform random drops → destination

imple random traffic generates complicated attractors

- TCP reacts to network traffic randomness
  - Jittery end-to-end delays



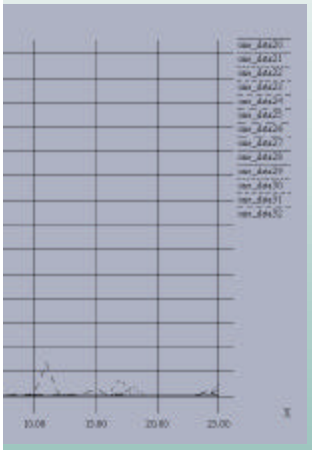
- Do not need chaos to generate complicated attractors
  - Poincare map of message delay vs. window size

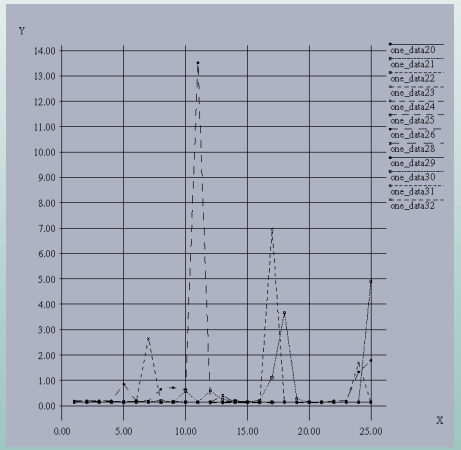


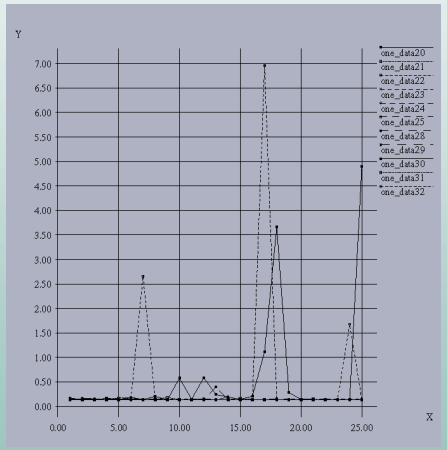
Jitter in End-to-End Delays
(wireless LAN - Internet measurements)

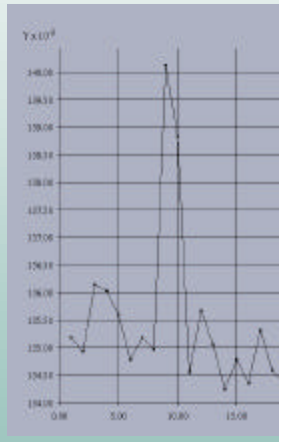
- Source: constant sized messages sent at regular intervals

ments from SC2001 booth to ORNL: Wireless network: 10K msgs - sent 25 times
n intervals - approximately every half hour – 13 measurements









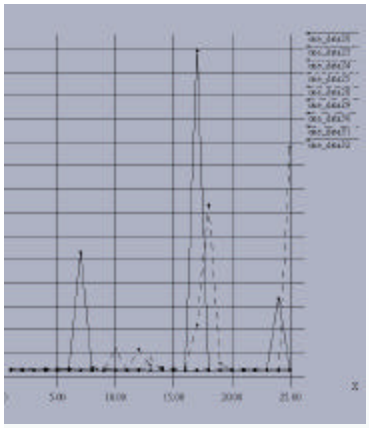
0-140 sec

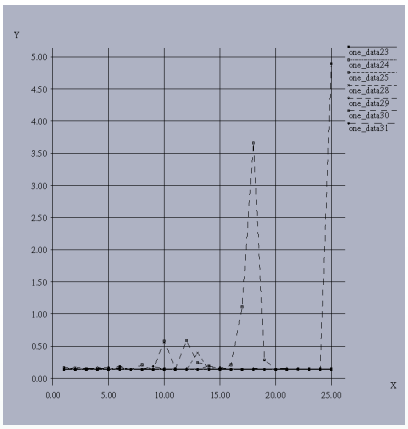
Scale: 0-14 sec

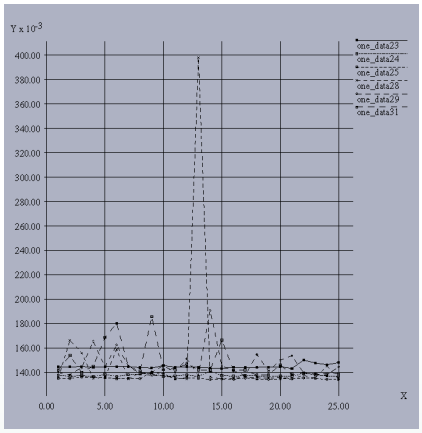
Scale: 0-7 sec

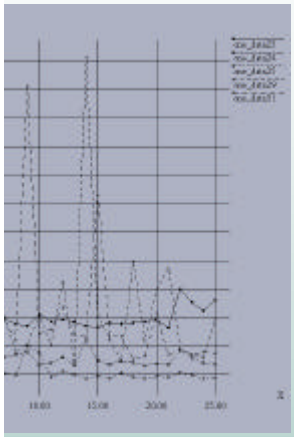
Scale: 0-150

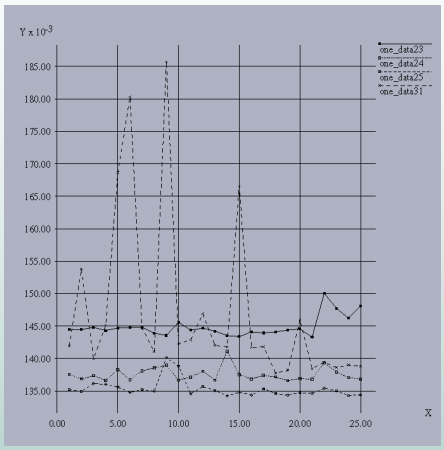
Measurements from SC2001 booth to ORNL: Wireless network

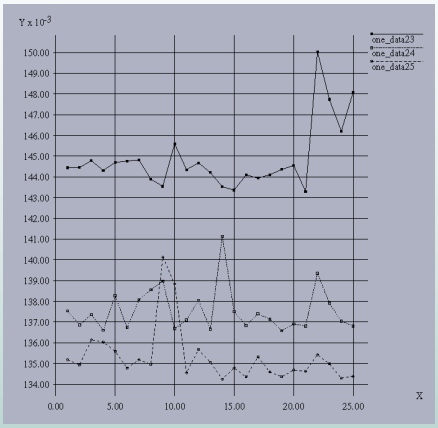


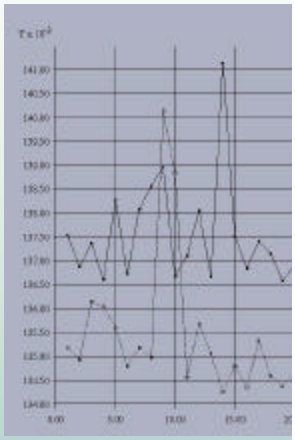


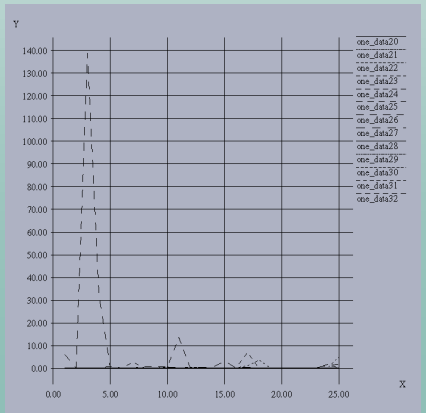


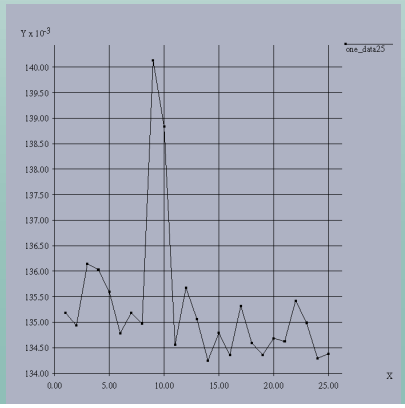










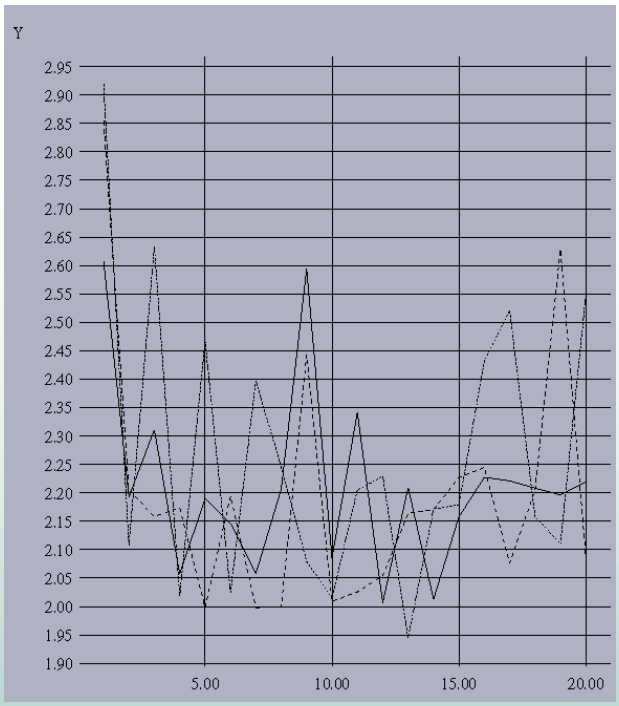


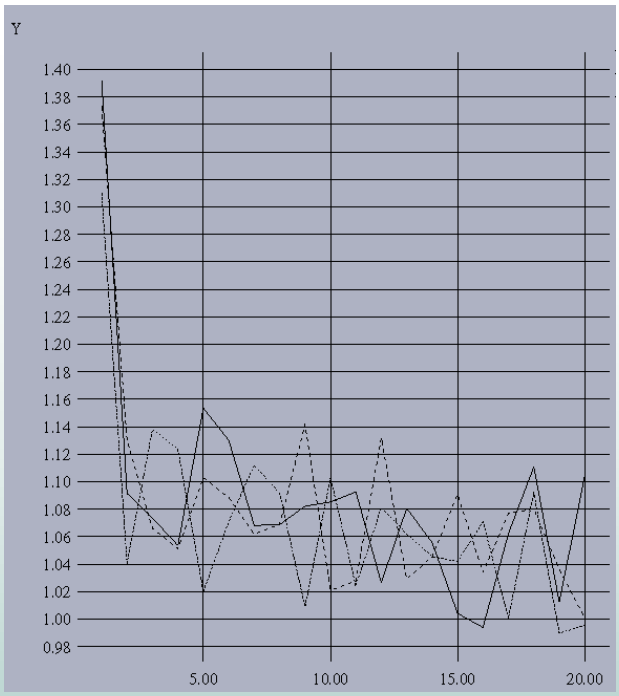
artment of Energy
e National Laboratory

UT-BATT

Internet Measurements from ORNL to OU (wired ~1000 mile Internet connection)

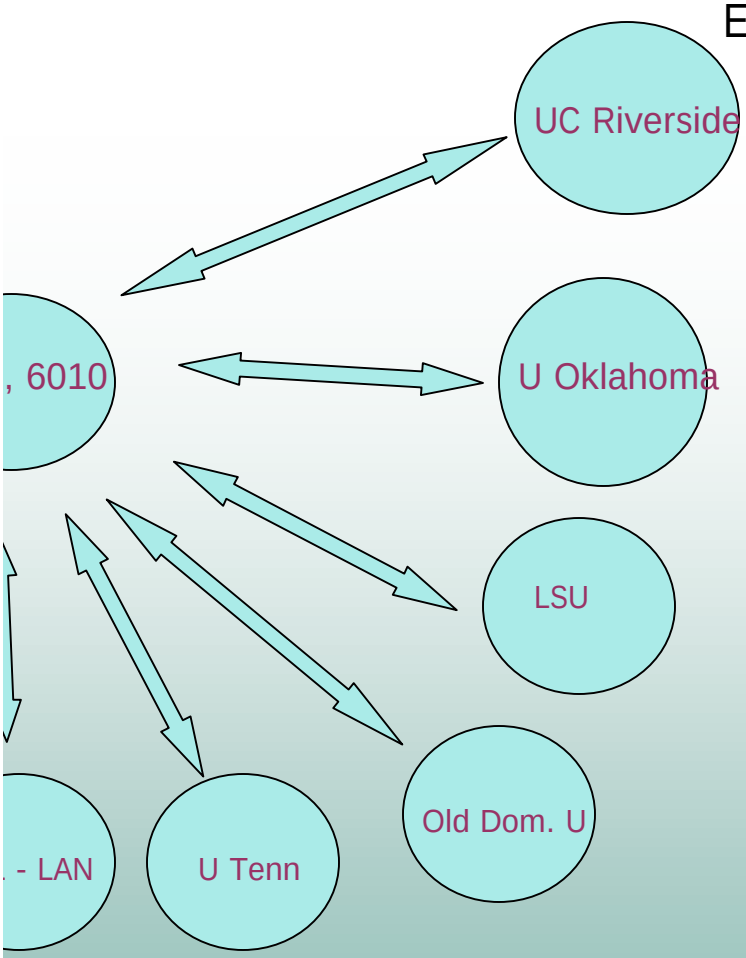
Time vs. end-to-end delay for fixed size messages sent regularly

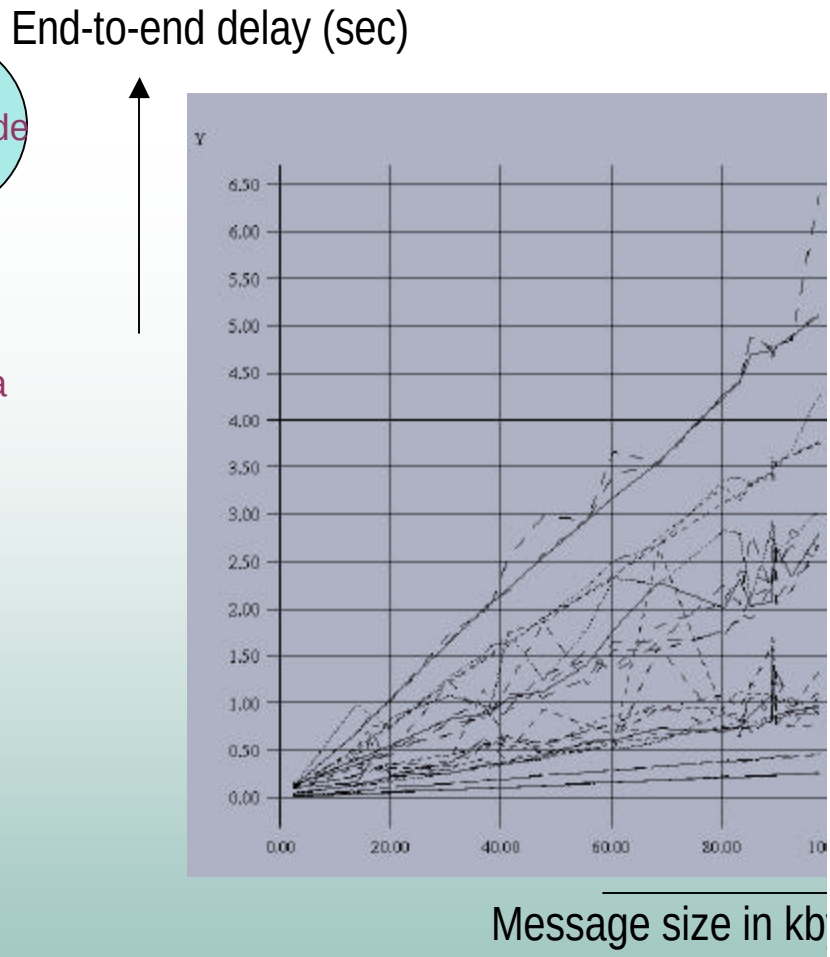




If messages are control signals to an end-instrument, the jitter can destabilize the control-loop

ork Measurements





TCP/IP end-to-end delivery times vs message size

End-to-End Delay Reduction

ernet Measurements:Four Daemons:

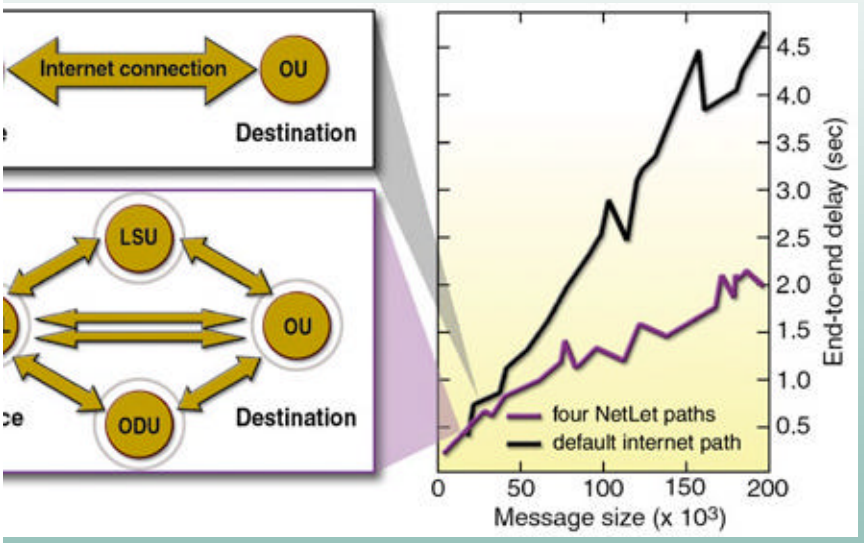
ır TCP streams: message optimally divided among

- Two parallel TCP streams
- Two streams via daemon routers

sults:

ırly readings in a week:

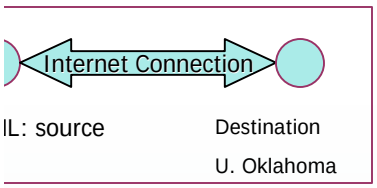
- Average 40% reduction in end-to-end delay
- Require no support from the network routers or service providers
- Daemons are user-level programs – can be executed using simple user accounts or on free telnet sites

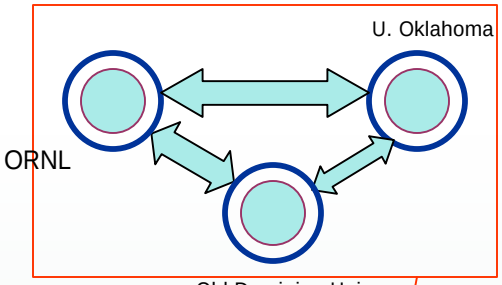


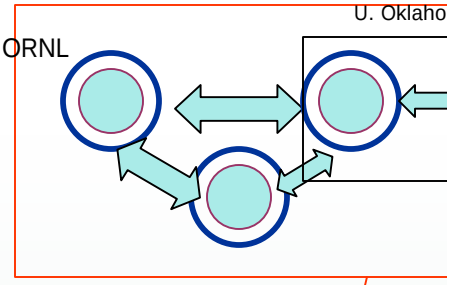
Ongoing Work:

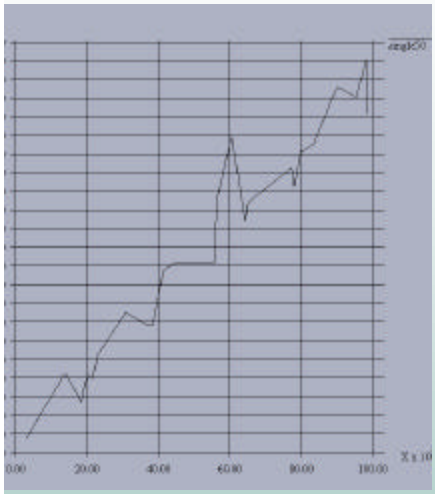
- Optimization of network measure
  - Statistical design of experiment
    - reduce measurement nodes a

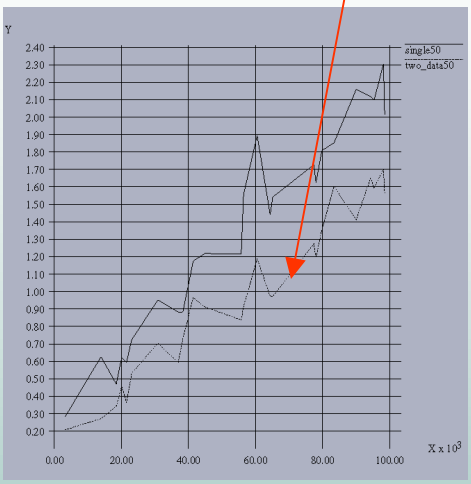
ernet Measurements: End-to-End Delay Dynamics Control

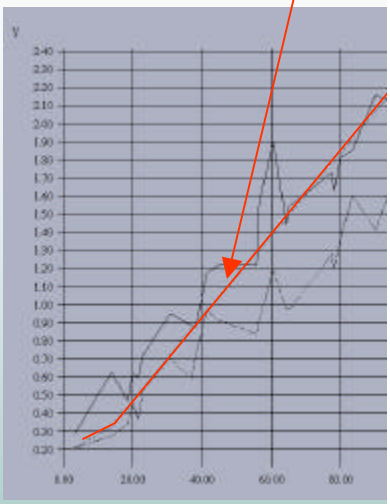












X-axis: message sizes (bytes)

Y-axis: end-to-end delay (sec)

mooth end-to-end delay

end-to-end delay using two-paths via daemons:
U, ORNL-ODU_OU
e output at destination

Funded by:

DARPA NMS Program

DOE High-performance n/w program

artment of Energy
e National Laboratory

UT-BATT

nclusions

- Networking Aspects
  - Study of end-to-end dynamics of transport mechanisms
    - Multiple paths, parallel TCP, TCP tuning and adaptation
    - Complex and chaotic regimes of transport methods
  - Develop method for controlling dynamics
    - End-filters, multiple throttles, adaptive performance tuning
- High-Performance Cooperative Computing Environments
  - Have Acute network performance needs
    - Not met by current operational networks – infrastructure and middleware
    - Will need targeted efforts; Will not be incidentally solved by other efforts
  - Provide supportive environments
    - Customized instruments can be installed to support
      - Avoid Congested parts
      - Aggregated bandwidths
      - Controlled dynamics