A Precision Infrastructure for Active Probing

Attila Pásztor[1,2]
Darryl Veitch[1]

1 - EMULab at the Department of Electrical & Electronic Engineering The University of Melbourne
2 - Ericsson Hungary R&D

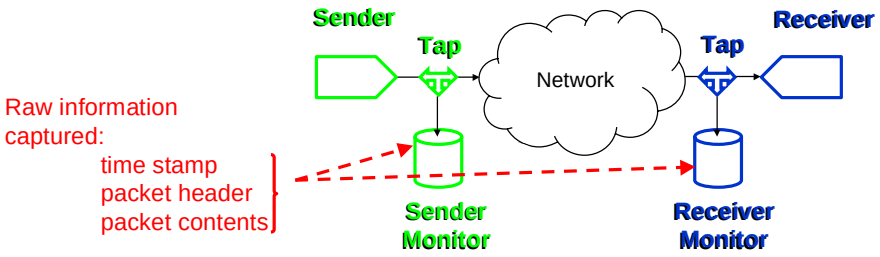
Introduction

1. **Active Measurement**
2. **Timing Control**
3. **A Precision Infrastructure for Active Probing**
4. **Early Experiments**

Active Measurement

Active measurement is the injection of artificial probe traffic and the measurement of its characteristics.

- \- invasive character
- \+ end to end nature
  - flexibility in probe stream design
  - measurement data volume reduction
  - avoidance of data privacy issues



3

Timing Control

- **Terminology**
  - resolution
  - offset $\Theta(t)$
  - skew γ
  - stability ~ $\omega(t)$

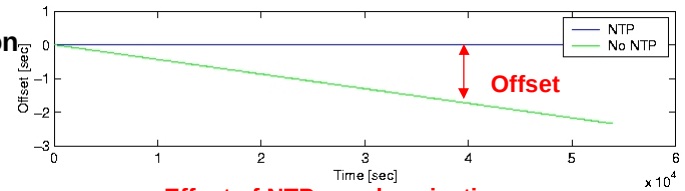
$$\Theta(t) = \gamma \, . \, t + \omega(t)$$

- **PC clocks**
  - SW clock - based on a periodic interrupt, triggered by a timer chip
  - using processor TSC register - µs resolution
  - typical skew 50 PPM, typical stability 0.1 PPM
- **Clock synchronisation**
  - NTP
  - GPS or other reliable reference clock source
- **Scheduling**
  - executing tasks on the desired time

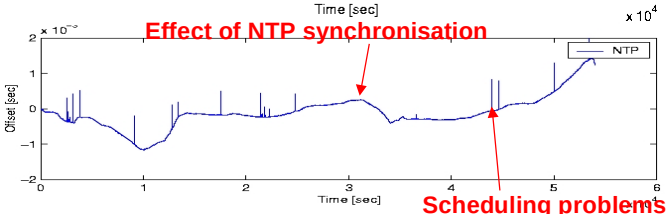
4

Timing Control

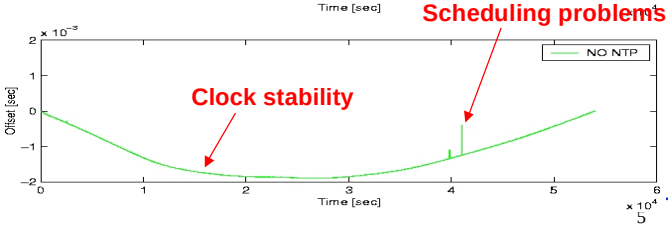
- Offset Comparison



- NTP close-up

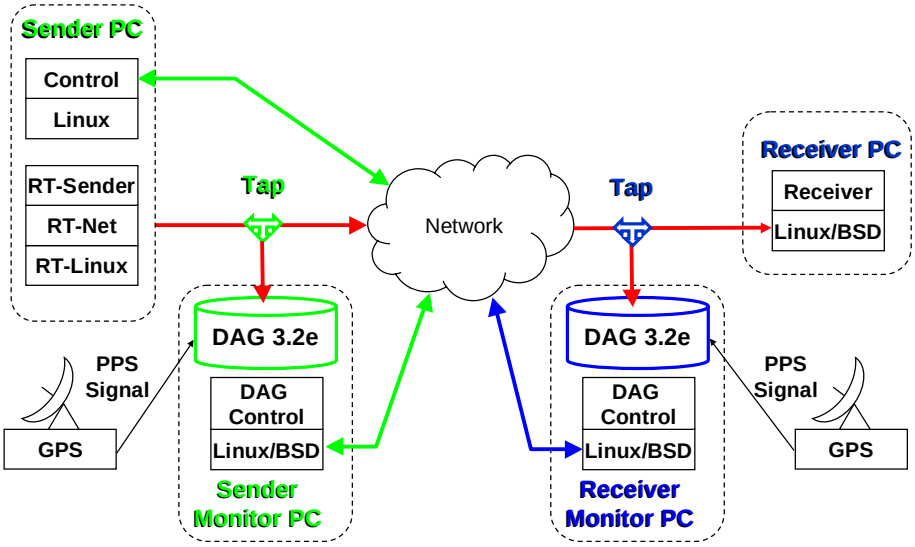


- NO-NTP corrected for the skew



5

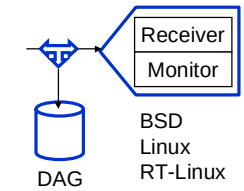
A Precision Infrastructure for Active Probing



6

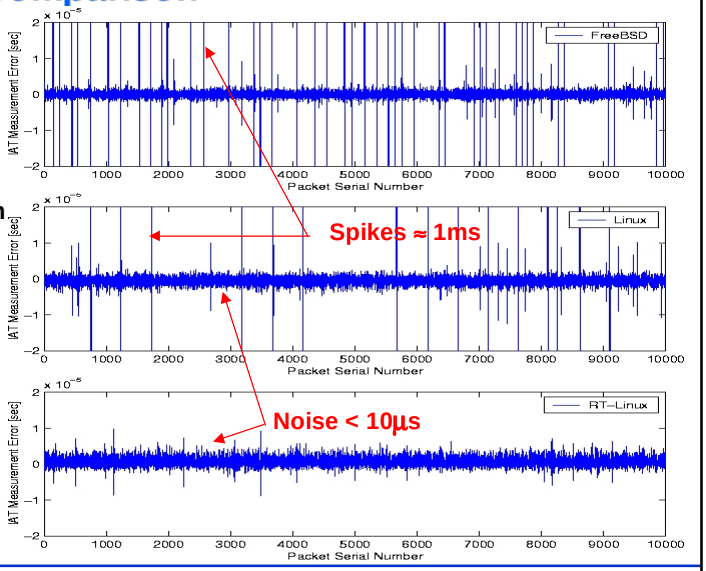
Monitoring Comparison

- **DAG used as a reference**
- **The measured packet inter-arrival time is compared**
- **Pseudo Poisson stream of 10000 UDP packets with mean inter-arrival time of 10ms**



DAG
Ref.monitor

BSD
Linux
RT-Linux



7

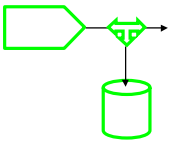
The Senders

- **The Sender defines and injects the active probe stream into the network**
  - the probe stream is defined by
    - **packet type (UDP,ICMP,..)**
    - **packet contents**
    - **sending process - pattern of inter-departure times and packet sizes**
- **Scheduling algorithms**
  - Active - unnecessary CPU load, resulting in scheduling problems
  - Passive - using `usleep` and `ualarm`, resolution problems
  - Hybrid - low CPU load, high resolution
- **The current version of the Sender**
  - is using hybrid scheduling
  - is implemented under BSD, Linux and RT-Linux
  - sends UDP packets with a serial number as payload

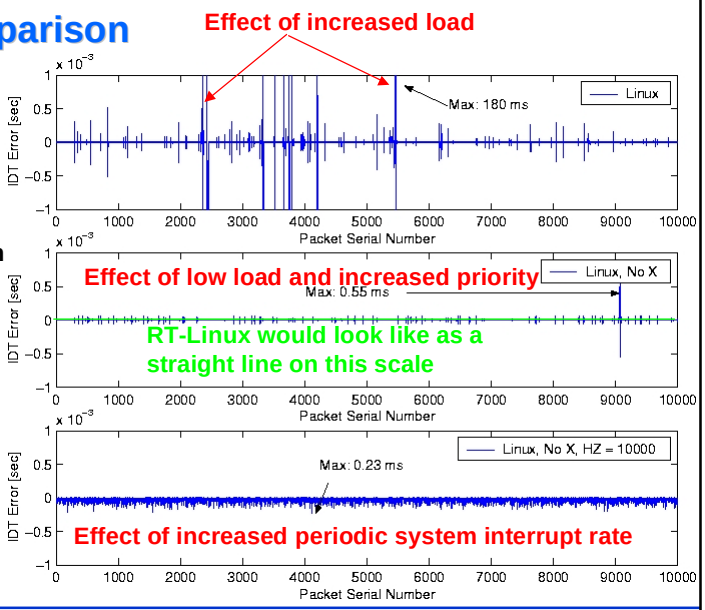
8

Sender Comparison

- DAG used as a monitor
- The target and measured packet inter-departure time is compared
- Pseudo Poisson stream of 10000 UDP packets with mean inter-arrival time of 10ms

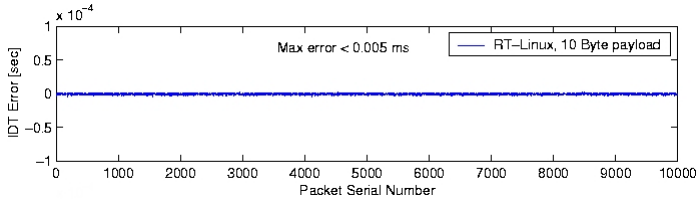


DAG Monitor



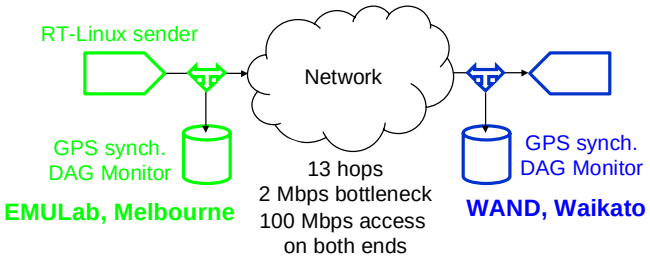
Sender Comparison

- RT-Linux



Early experiments

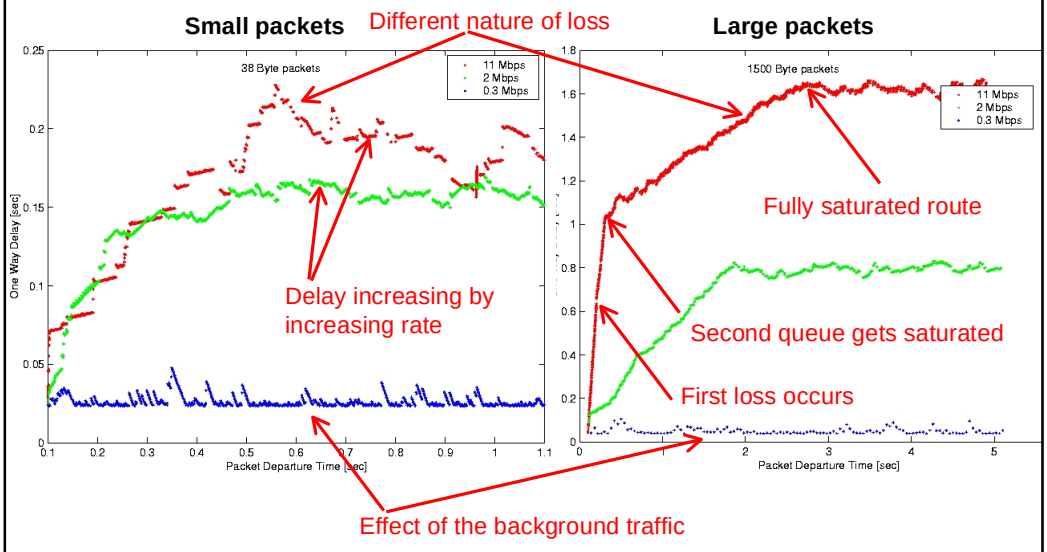
- **Configuration**



- **Experiment I.**
  - Sending short periodic streams of *small* and *large* packets with rates well below, app. around and well above the bottleneck rate
- **Experiment II.**
  - Comparison of a periodic and a simple designer probe stream

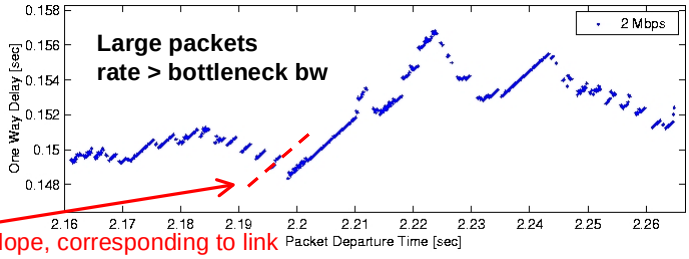
11

Experiment I.

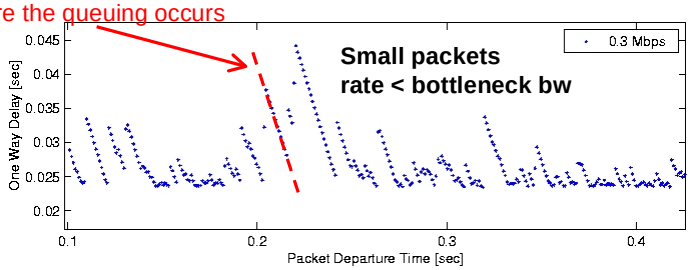


12

Experiment I.



Characteristic slope, corresponding to link bandwidth where the queuing occurs

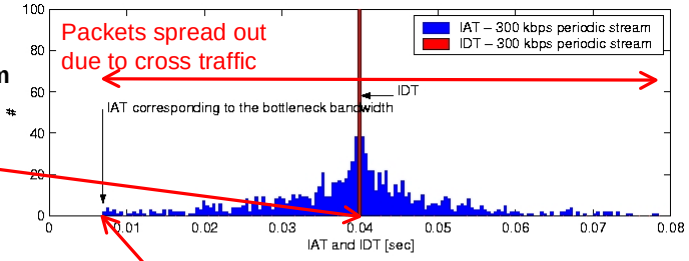


13

Experiment II.

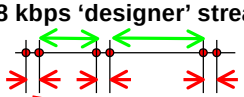
300 kbps periodic stream

IDT corresponding to the period

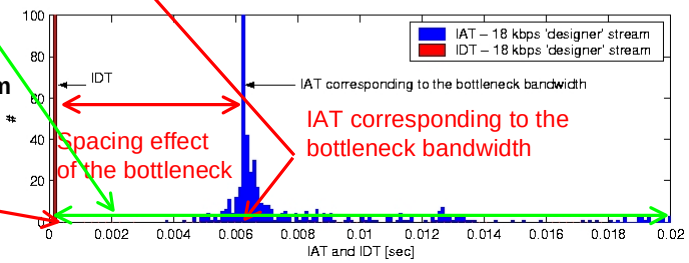


Poisson distributed inter pair IDT spread out

18 kbps 'designer' stream



IDT corresponding to the intra pair period



14

Conclusions

- **A highly accurate active probing infrastructure was developed**
  - reliable monitoring
  - reliable transmission of high resolution probe streams
  - accuracy << 1ms
- **Comparison of different Unix on PC solutions**
  - RT-Linux clearly superior, scheduling errors ≈ 10µs
  - BSD and Linux - errors in the range of 1ms
- **Network based clock synchronization**
  - produces undesirable effects in the sub-millisecond range - NOT RECOMMENDED
- **Preliminary experiments**
  - system accuracy demonstrated
  - high resolution histograms - leading to new means of measuring link rates

15