Detection and Mitigation of Impairments for Real-Time Multimedia Applications

Soshant Bali

Ph.D. Final Exam, December 3, 2007

Research Questions

- Can we
  - define RTM impairments?
  - detect RTM impairments?
  - detect causes of RTM impairments?
  - mitigate RTM impairments?

Main Contributions (I)

- Methods to detect RTM impairments
- Methods to detect causes of impairments
  - Congestion
    - Heuristics-based
  - Route changes
    - Heuristics-based
    - Model-based
      - Optimal model-based route change detector – parameter aware (PAD): provides performance bound
      - Analysis to predict performance of PAD
      - Practical model-based route change detector – parameter unaware (PUD)
      - Compare performance of heuristics-based, PUD and ideal
  - Evaluation using Internet measurements

Main Contributions (II)

- Methods to mitigate RTM impairments
  - Discovery that proportional fair (PF) scheduler induces RTM impairments
  - New scheduler that mitigates impairments
  - New alpha initialization strategy

Relevance of this Research

- Detect RTM impairments
  - Impairments QoS metric for SLAs
- Detect causes of RTM impairments
  - Fault/state detection for ISPs
  - Routing for overlay/underlay networks
  - Improved Internet tomography
  - Next-steps in signaling
  - Improved minimum RTT estimation
  - TCP throughput improvement
- Mitigate RTM impairments
  - Robust schedulers for wireless networks

Outline

- Mitigating RTM impairments
  - PF scheduler starvation problem
  - Robust scheduling
- Detecting impairments
- Detecting causes of impairments
  - Heuristic methods: congestion, route changes
  - Model-based methods: route changes

Discussed in Ph.D. proposal

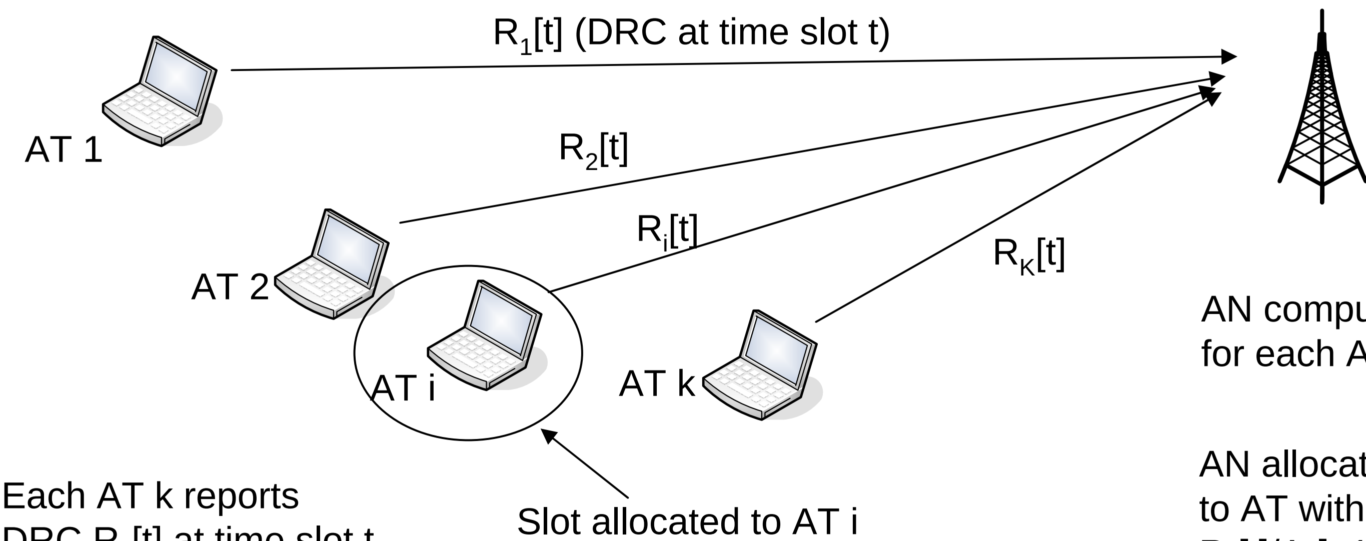
PF Scheduler

- PF Scheduler
  - Channel aware, downlink scheduler
  - Maximizes system throughput
  - Long-term fairness
  - Widely deployed: EVDO, HSDPA
  - Schedule ATs with good channel conditions
    - Each AT k reports achievable rate $R_k(t)$ in slot t
    - Scheduler calculates average rates $A_k(t)$

$$A_k[t] = \begin{cases} (1-\alpha) A_k[t-1] + \alpha R_k[t] & \text{if k is scheduled in slot t} \\ (1-\alpha) A_k[t-1] & \text{if k is not scheduled in slot t} \end{cases}$$

    - Schedule AT with maximum $R_k(t)/ A_k(t-1)$

How PF scheduler works



AN computes $R_k[t]/A_k[t-1]$ for each AT k

Each AT k reports DRC $R_k[t]$ at time slot t

AN allocates time slot to AT with maximum $R_k[t]/A_k[t-1]$, say AT i

Exponential weighted average throughput to each AT

$$A_k[t] = \begin{cases} (1-\alpha) A_k[t-1] + \alpha R_k[t] & \text{if k is scheduled in slot t} \\ (1-\alpha) A_k[t-1] & \text{if k is not scheduled in slot t} \end{cases}$$

$A_i[t]$ of AT i updates as

$$A_i[t] = (1-\alpha) A_i[t-1] + \alpha R_i[t]$$

$A_k[t]$ of all other ATs k updated

$$A_k[t] = (1-\alpha) A_k[t-1]$$

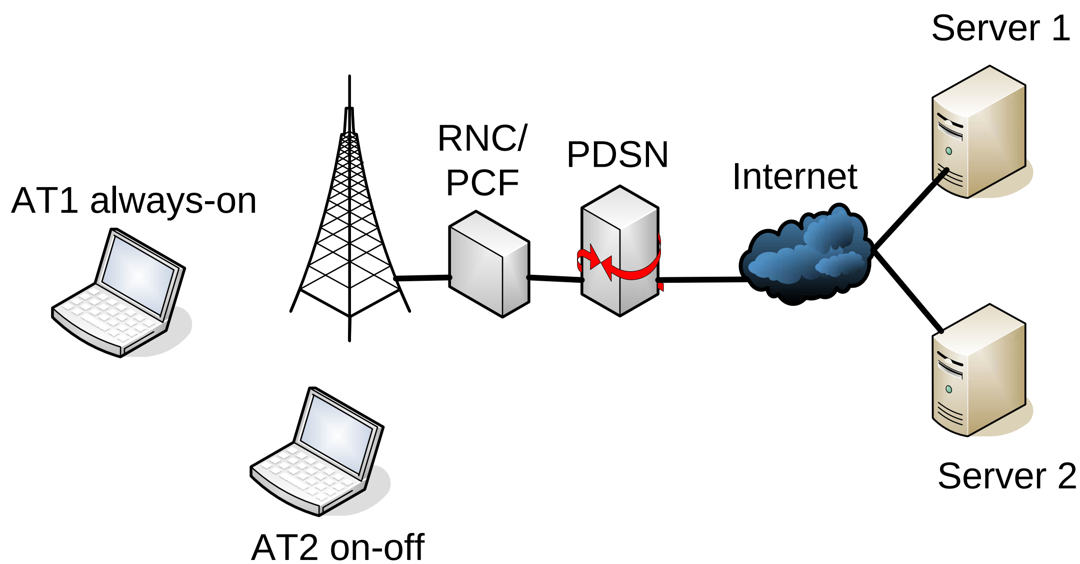
Schedule AT that has its better than average conditions
i.e., schedule AT with maximum R/A

PF Scheduler Starvation

- PF design assumes infinite backlog
  - Traffic commonly on-off, e.g., web browsing
- Problem: on-off traffic causes starvation
  - When off, no slots allocated to that AT
  - Average decays when no slots allocated
  - When on after long off, average is very low
  - AT that restarts has highest R/A amongst all ATs (low A)
  - AT that restarts gets all slots until A increases
  - This starves other ATs
- PF widely deployed and can be easily corrupted
  - Deliberately (attacks using burst UDP)
  - Accidentally (web browsing)

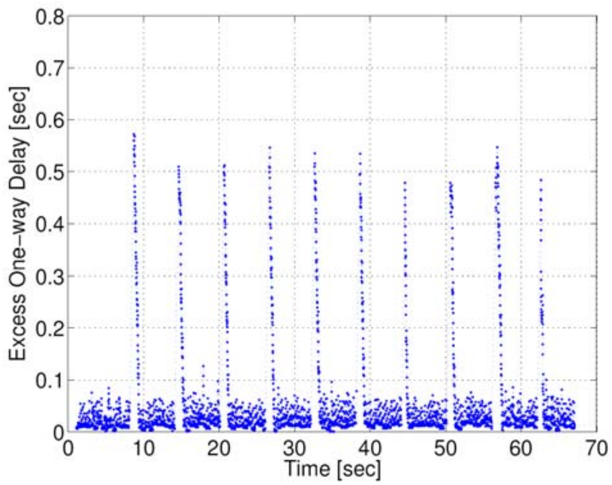
Measurement Setup

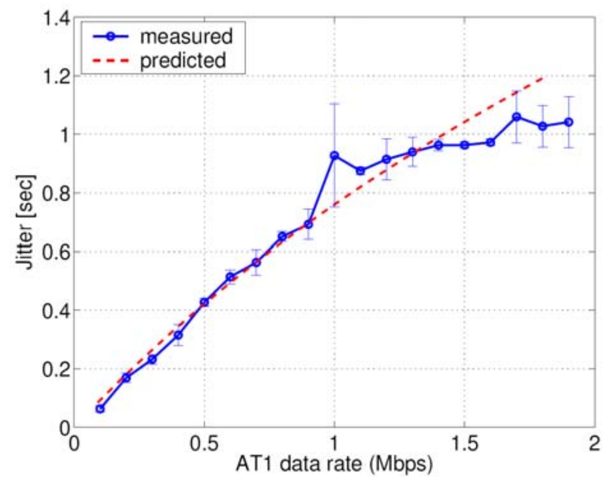
- Experiments in deployed network and in laboratory
  - No cross traffic in laboratory



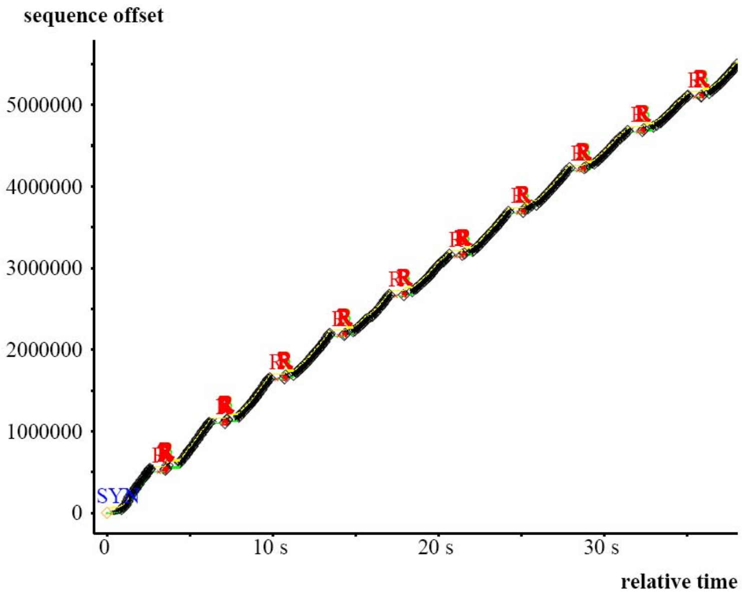
Scheduler-Induced Jitter

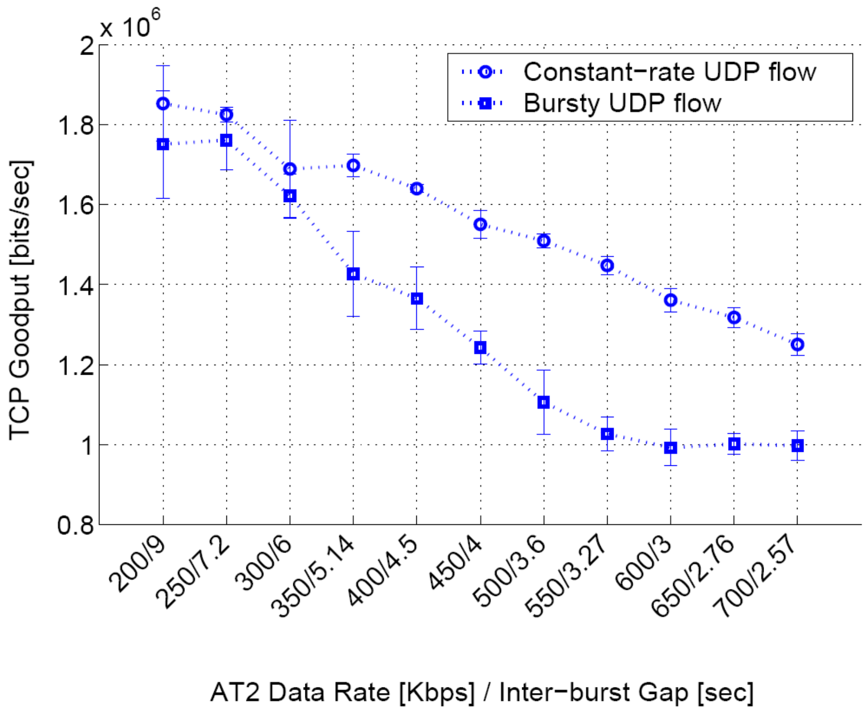
- AT1: cbr traffic; AT2: 250 1500B pkts every 6 sec
- Increase in delay a function of AT1s rate
  - Assume DRCs are constant
  - If $A1_T = \beta 1_T R1$ and $A2_T = \beta 2_T R2$, then jitter=time until $\frac{R1}{A1_{t-1}} > \frac{R2}{A2_{t-1}}$
  - Predicted jitter $J = \left\lceil \frac{\log\left(\frac{1}{1+\beta 1_T - \beta 2_T}\right)}{\log(1-\alpha)} \right\rceil$
  - Predicted jitter matches measured jitter when $\beta 2_T = 0$





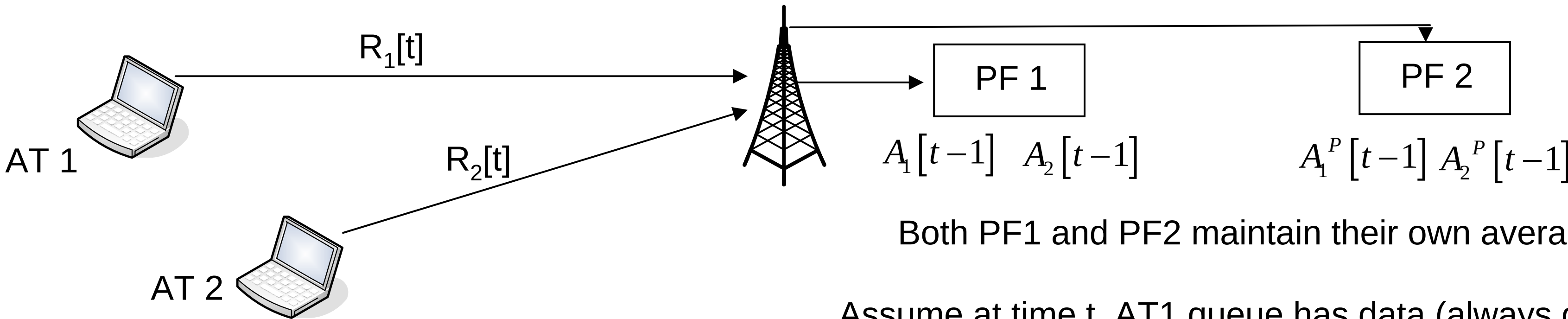
Measurement Results





AT1 downloads 20MB file. AT2 receives cbr or bursty traffic. A burst has 150 pkts of size 1500 Bytes each.

Solution: parallel PF scheduler



Both PF1 and PF2 maintain their own averages

Assume at time t, AT1 queue has data (always on) and AT2 queue is empty (in off state)

Compute $R_k[t]/A_k[t-1]$ for AT k=1, since there is no data for AT2. Final scheduling decided by PF1

Compute $R_k[t]/A_k^P[t-1]$ for each AT k. PF2 does not look at the Queues.

Update $A_k[t]$ for all k

Update $A_k^P[t]$ for all k

At time t+M, AT2 queue receives data for AT2

$A_k[t+M-1] = A_k^P[t+M-1]$

Compute $R_k[t]/A_k[t-1]$ for all k. AT with highest ratio gets slot.

Summary:-

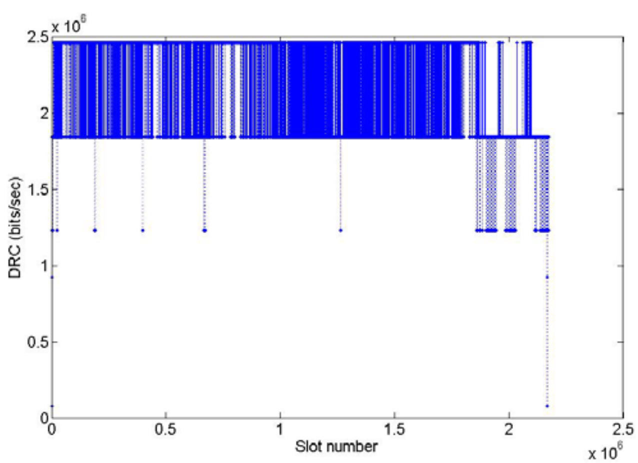
- PF1 decides final scheduling
- PF2 only virtual scheduling
- PF1 aware of queue size
- PF2 unaware of queue size
- When on after off, copy averages from PF2 to PF1

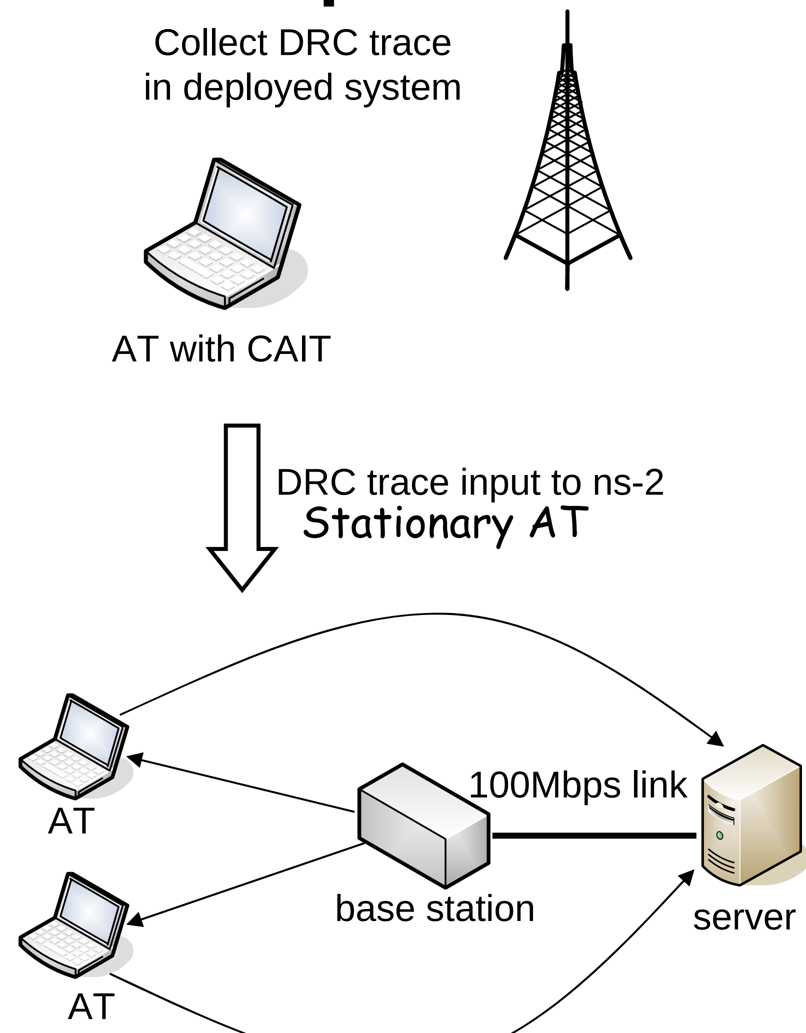
Robust Scheduling

- Adaptive alpha initialization
  - After long inactivity, AT dormant
  - No SINR reporting in dormant mode
  - Parallel PF cannot work
  - Initialize alpha for faster convergence of average
  - Shortens starvation duration

Simulation setup

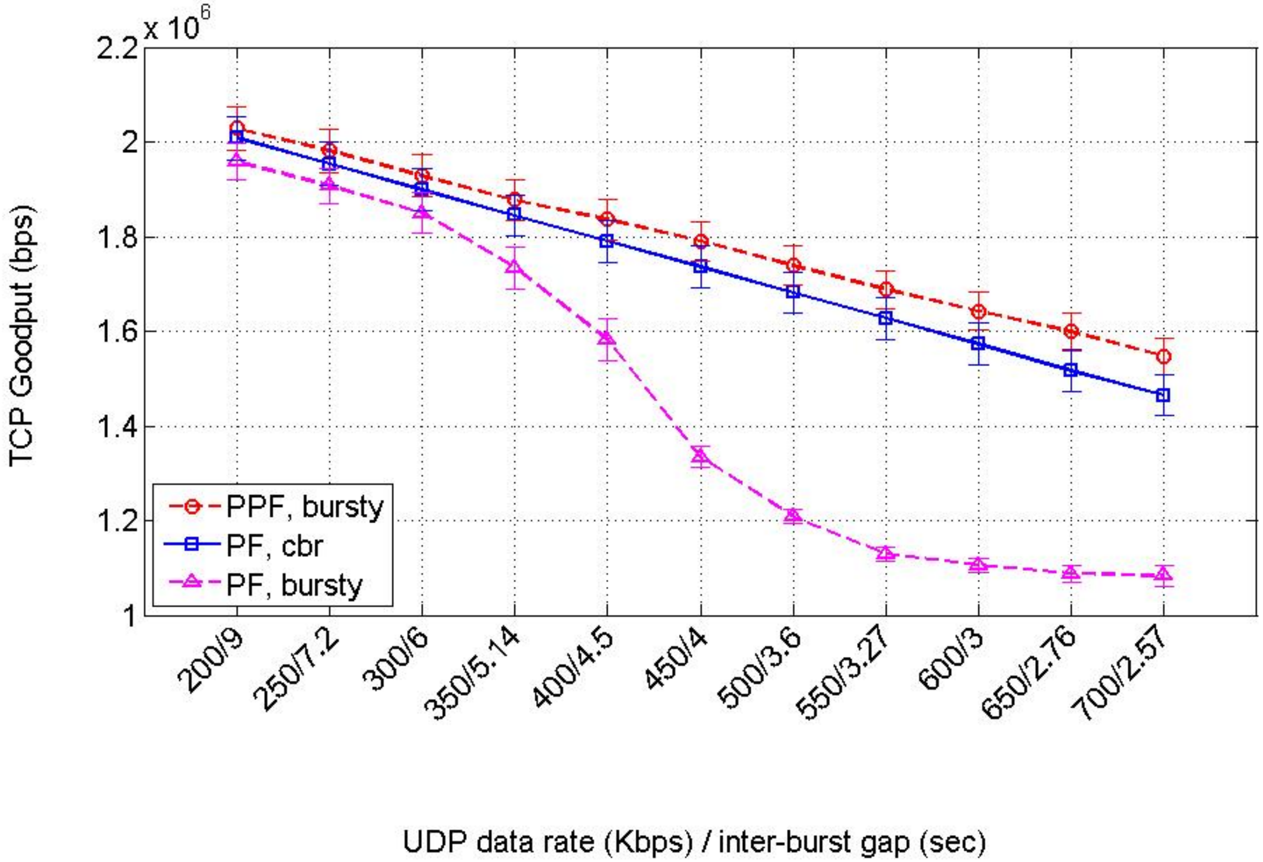
- Collect stationary user DRC trace in deployed system using CDMA air interface tester (CAIT)
- DRC trace input to ns-2
- Server-base station 100Mbps
- Base-station to AT DRC variable (from trace)
  - Loss probability = 0
  - RLP not implemented (not needed)
- High bandwidth link from AT to server





ns-2 model

Parallel PF: simulation results



Model-Based Approach

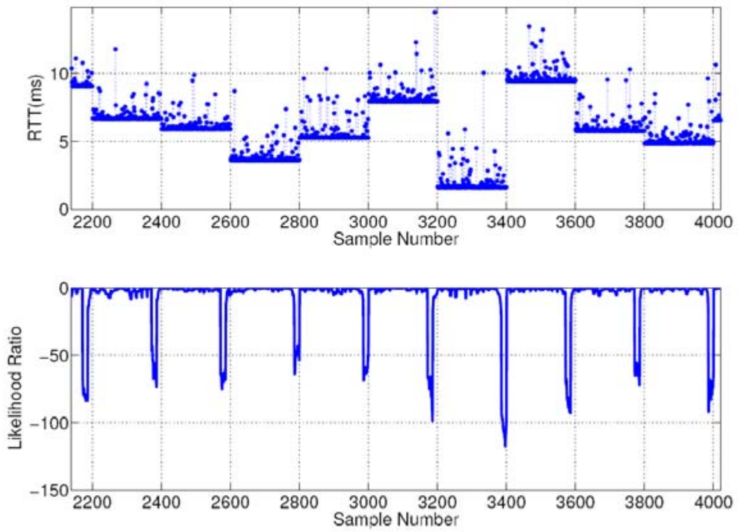
- Model-based v/s heuristic
  - Predictable performance
  - Quantifiable performance tuning
  - Better performance?
  - Provides theoretical performance bound
- Model-based detection
  - RTTs i.i.d. samples from Gamma distribution
  - Hypothesis test
    - H_0: All n samples from Gamma: $\alpha_0, \beta_0, \gamma_0$
    - H_1: First $\lfloor n/2 \rfloor$ from Gamma: $\alpha_1, \beta_1, \gamma_1$ and next $\lceil n/2 \rceil$ from Gamma: $\alpha_2, \beta_2, \gamma_2$
    - Find likelihood ratio L
    - If $L > \lambda$ then H_0 true, else H_1 true

Parameter-Unaware Detector

- Assume RTTs modeled with Gamma PDF

$$f_T(t \mid \alpha,\beta,\gamma) = \begin{cases} \frac{1}{\Gamma(\alpha)\beta^{\alpha}}(t-\gamma)^{(\alpha-1)} e^{-\frac{(t-\gamma)}{\beta}} & \gamma \le t < \infty \\ 0 & \text{otherwise} \end{cases}$$

- Given n samples, estimate parameters using: -
  - All n samples: $\hat{\alpha}_0, \hat{\beta}_0, \hat{\gamma}_0$
  - First $\lfloor n/2 \rfloor$ samples: $\hat{\alpha}_1, \hat{\beta}_1, \hat{\gamma}_1$
  - Last $\lceil n/2 \rceil$ samples: $\hat{\alpha}_2, \hat{\beta}_2, \hat{\gamma}_2$



- find L

$$L = Log \frac{\prod_{i=1}^{n} f_T\left(t = t_i \mid \hat{\alpha}_0, \hat{\beta}_0, \hat{\gamma}_0\right)}{\prod_{j=1}^{\lfloor n/2 \rfloor} f_T\left(t = t_j \mid \hat{\alpha}_1, \hat{\beta}_1, \hat{\gamma}_1\right) \prod_{k=\lfloor n/2 \rfloor+1}^{n} f_T\left(t = t_k \mid \hat{\alpha}_2, \hat{\beta}_2, \hat{\gamma}_2\right)}$$

- If $L > \lambda$ then H_0 true, otherwise H_1 true

Parameter-Aware Detector

- Assume RTTs modeled with Gamma PDF

$$f_T(t \mid \alpha, \beta, \gamma) = \begin{cases} \frac{1}{\Gamma(\alpha)\beta^{\alpha}} (t-\gamma)^{(\alpha-1)} e^{-\frac{(t-\gamma)}{\beta}} & \gamma \le t < \infty \\ 0 & \text{otherwise} \end{cases}$$

- Given n samples, and 9 parameters, find L

$$L = Log \frac{\prod_{i=1}^{n} f_T(t = t_i \mid \alpha_0, \beta_0, \gamma_0)}{\prod_{j=1}^{\lfloor n/2 \rfloor} f_T(t = t_j \mid \alpha_1, \beta_1, \gamma_1) \prod_{k=\lfloor n/2 \rfloor + 1}^{n} f_T(t = t_k \mid \alpha_2, \beta_2, \gamma_2)}$$

- If $L > \lambda$ then H_0 true, otherwise H_1 true
- Observations
  - Not practical: prior knowledge of parameters
  - Optimal detector in likelihood ratio sense

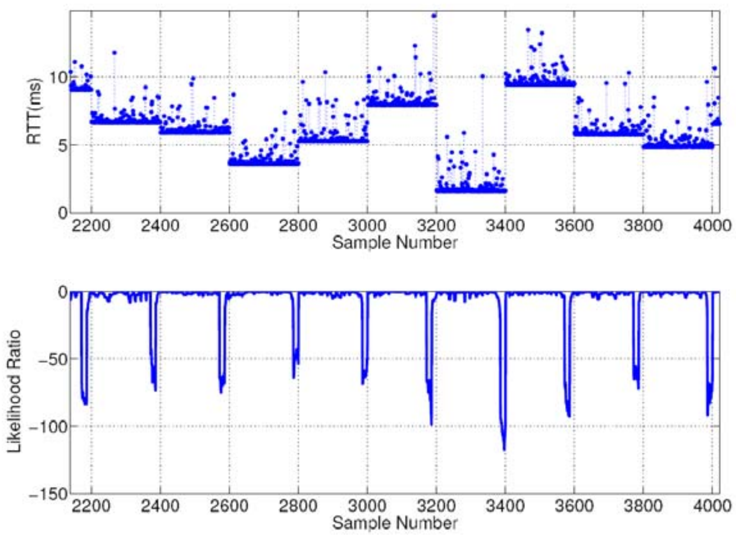
Performance Metrics

- Useful to know the functions

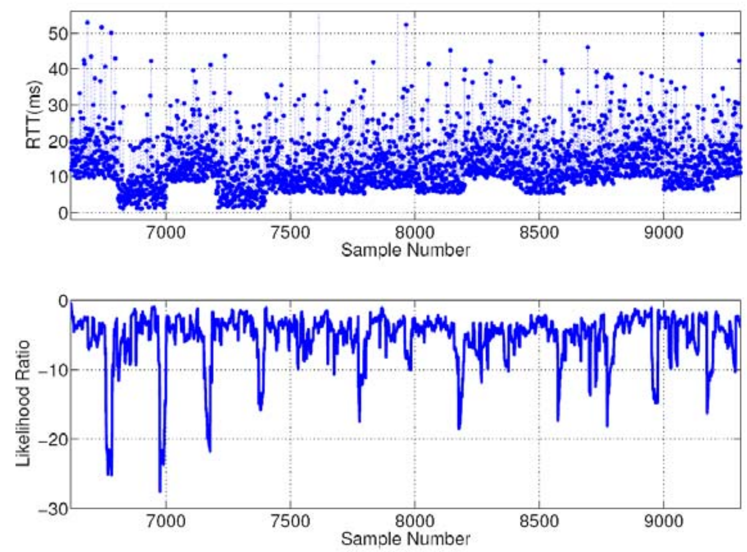
$$P_D = f^D(\lambda, n, \alpha_0, \beta_0, \gamma_0, \alpha_1, \beta_1, \gamma_1, \alpha_2, \beta_2, \gamma_2)$$

$$P_F = f^F(\lambda, n, \alpha_0, \beta_0, \gamma_0, \alpha_1, \beta_1, \gamma_1, \alpha_2, \beta_2, \gamma_2)$$

$$\lambda = f^\lambda(P_D, P_F, n, \alpha_0, \beta_0, \gamma_0, \alpha_1, \beta_1, \gamma_1, \alpha_2, \beta_2, \gamma_2)$$



$\alpha = 0.12, \beta = 1.99, \mathrm{n} = 30$



$\alpha = 1.2, \beta = 6, \mathrm{n} = 50$

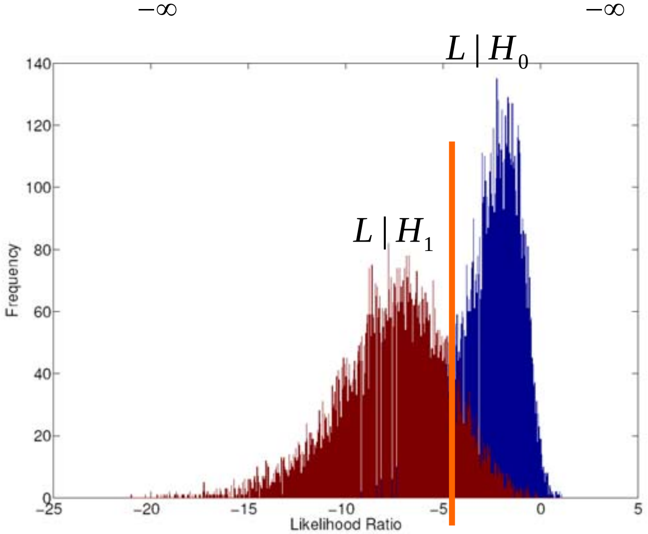
Performance Metrics (II)

- To derive expressions for P_D, P_F and λ , need PDF of $L|H_0$ and $L|H_1$

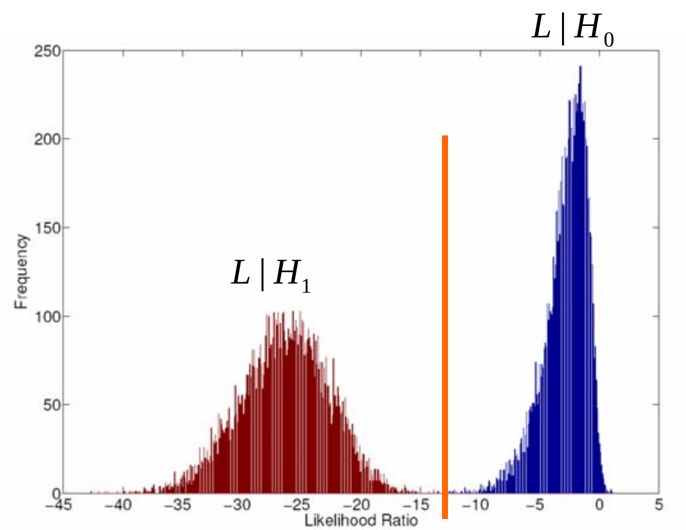
$$f_{L|H_0}(l) = f^{L|H_0}\left(n, \alpha_0, \beta_0, \gamma_0, \alpha_1, \beta_1, \gamma_1, \alpha_2, \beta_2, \gamma_2\right)$$
$$f_{L|H_1}(l) = f^{L|H_1}\left(n, \alpha_0, \beta_0, \gamma_0, \alpha_1, \beta_1, \gamma_1, \alpha_2, \beta_2, \gamma_2\right)$$

- Then,

$$P_D = \int_{-\infty}^{\lambda} f_{L|H_1}(l)\, dl \qquad P_F = \int_{-\infty}^{\lambda} f_{L|H_0}(l)\, dl$$



$\alpha = 1.2, \beta = 6, \Delta T = 1\text{ms}$



$\alpha = 1.2, \beta = 6, \Delta T = 5\text{ms}$

PAD Analysis

- What is PDF of $L|H_0$ and $L|H_1$?
  - Difficult: find first two moments instead
  - Assume $L|H_0$ and $L|H_1$: Gaussian PDF
- Parameter subspaces
  - H_0 true
    - If $\gamma_0 \geq \mathrm{Max}(\gamma_1, \gamma_2)$ *L-finite* space
    - If $\gamma_0 < \mathrm{Max}(\gamma_1, \gamma_2)$ *L-infinite* space
    - $P_{LF|H0}$ Equations 3.7, 3.8, 3.9
  - H_1 true
    - If $\gamma_0 \leq \mathrm{Min}(\gamma_1, \gamma_2)$ *L-finite* space
    - If $\gamma_0 > \mathrm{Min}(\gamma_1, \gamma_2)$ *L-infinite* space
    - $P_{LF|H1}$ Equations 3.45, 3.46, 3.47
- Expressions for first two moments
  - *L-finite:* $E[L|H_0]$ Eq. 3.16; $E[L^2|H_0]$ Eq.3.35; $E[L|H_1]$ Eq. 3.49; $E[L^2|H_1]$ Eq.3.71
  - *L-infinite:* $E[L|H_0]$ Eq. 3.40; $E[L^2|H_0]$ Eq.3.42; $E[L|H_1]$ Eq. 3.73; $E[L^2|H_1]$ Eq.3.74

PAD Analysis (II)

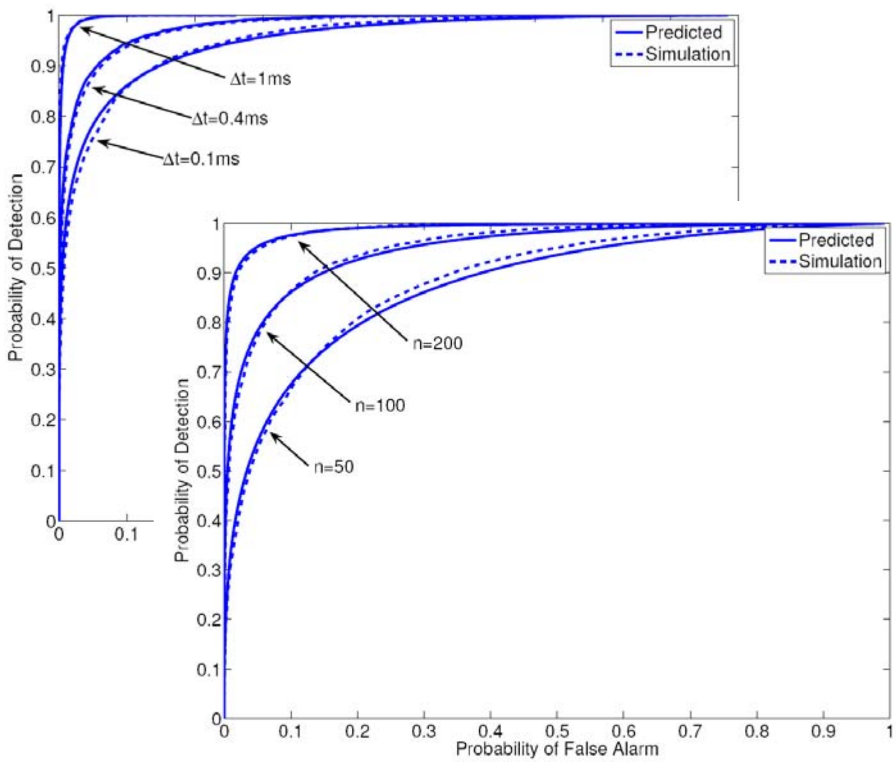
- Assume $L|H_0$ and $L|H_1$ are Gaussian RVs
- Then,

$$P_D = P_{LF|H_1} \frac{1 + erf\left(\lambda - \mu_{L|H1} \big/ \sigma_{L|H1}\sqrt{2}\right)}{2} + \left(1 - P_{LF|H1}\right)$$

$$P_F = P_{LF|H0} \frac{1 + erf\left(\lambda - \mu_{L|H0} \big/ \sigma_{L|H0}\sqrt{2}\right)}{2}$$

PAD Analysis: Validation

$\alpha = 2, \beta = 4, n = 100, \beta_2 = \beta + 1$

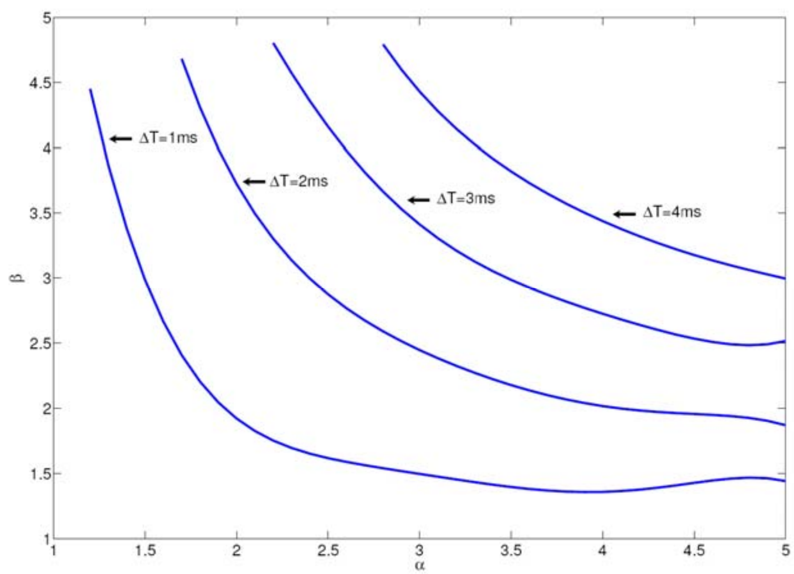


$\alpha = 2, \beta = 4, \Delta T = 0.1ms, \beta_2 = \beta + 1$

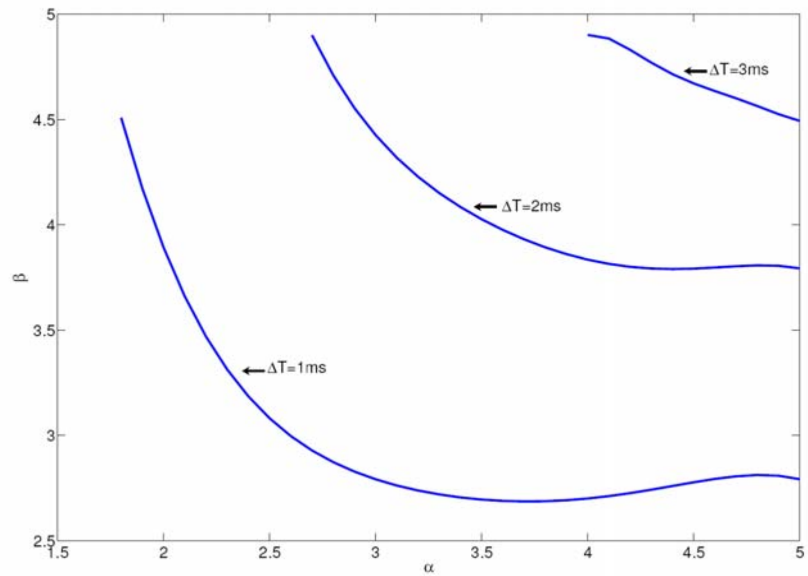
- Simulation setup
  - 10,000 samples of $L|H_0$
  - 10,000 samples of $L|H_1$
  - Vary threshold over the entire range
  - Find P_D and P_F for each value of threshold
- Result: Predicted ROCs match simulations
  - Moments separately validated
  - Validates analysis
  - Validates Gaussian assumption

PAD Acceptable Performance Region

- What are the parameter values for which PAD has acceptable performance?
- Acceptable performance region
  - Parameter space for which $P_D \geq 0.999, P_F \leq 0.001$



$n = 100$



$n = 300$

PUD Analysis

- Difficult
  - We need $f_{L|H_0}(l) = f^{L|H_0}\left(n, \alpha_0, \beta_0, \gamma_0, \alpha_1, \beta_1, \gamma_1, \alpha_2, \beta_2, \gamma_2\right)$

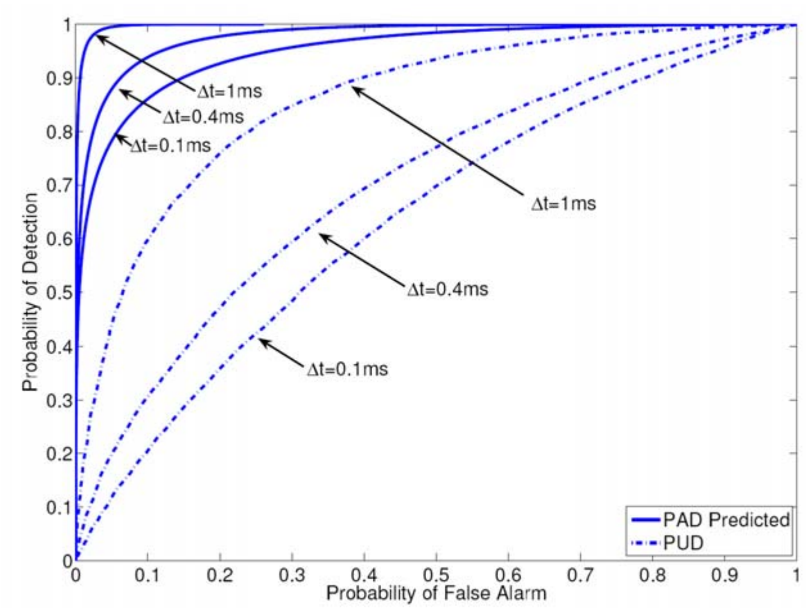
    $f_{L|H_1}(l) = f^{L|H_1}\left(n, \alpha_0, \beta_0, \gamma_0, \alpha_1, \beta_1, \gamma_1, \alpha_2, \beta_2, \gamma_2\right)$
  - $$L = Log \frac{\prod_{i=1}^{n} f_T\left(t = t_i \mid \hat{\alpha}_0, \hat{\beta}_0, \hat{\gamma}_0\right)}{\prod_{j=1}^{\lfloor n/2 \rfloor} f_T\left(t = t_j \mid \hat{\alpha}_1, \hat{\beta}_1, \hat{\gamma}_1\right) \prod_{k=\lfloor n/2 \rfloor+1}^{n} f_T\left(t = t_k \mid \hat{\alpha}_2, \hat{\beta}_2, \hat{\gamma}_2\right)}$$
  - Here $\hat{\alpha}_0, \hat{\beta}_0, \hat{\gamma}_0, \hat{\alpha}_1, \hat{\beta}_1, \hat{\gamma}_1, \hat{\alpha}_2, \hat{\beta}_2, \hat{\gamma}_2$ are random variables
  - Value of $\hat{\alpha}_0, \hat{\beta}_0, \hat{\gamma}_0$ correlated with $\hat{\alpha}_1, \hat{\beta}_1, \hat{\gamma}_1, \hat{\alpha}_2, \hat{\beta}_2, \hat{\gamma}_2$

PUD Simulation Methodology

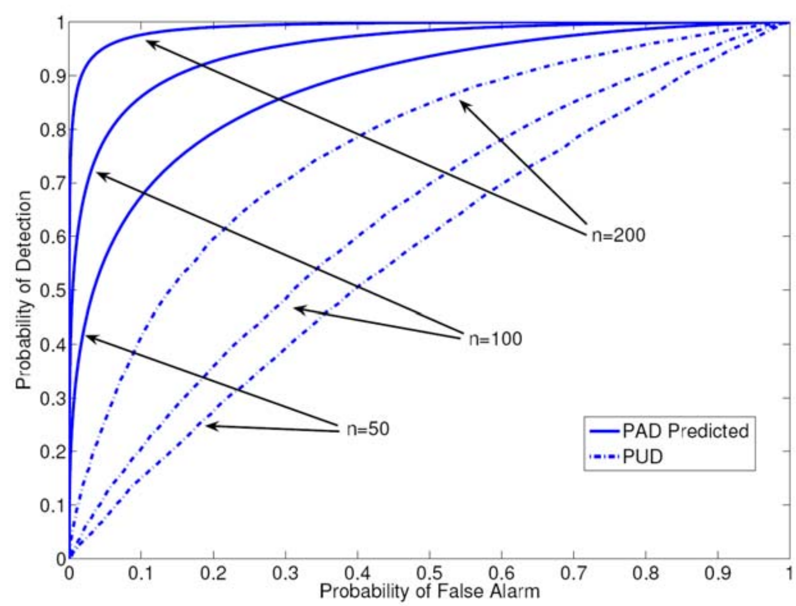
- Generate 10,000 windows of n samples
  - Generate samples using $\alpha_0, \beta_0, \gamma_0$
  - Apply PUD to each of the 10,000 windows
  - Gives 10,000 samples of $L|H_0$
- Generate 10,000 windows of n samples
  - First $\lfloor n/2 \rfloor$ samples using $\alpha_1, \beta_1, \gamma_1$
  - Next $\lceil n/2 \rceil$ samples using $\alpha_2, \beta_2, \gamma_2$
  - Apply PUD to each of the 10,000 windows
  - Gives 10,000 samples of $L|H_1$
- P_D and P_F can be estimated from distribution of $L|H_0$ and $L|H_1$

PUD ROCs

- PAD performance better than PUD

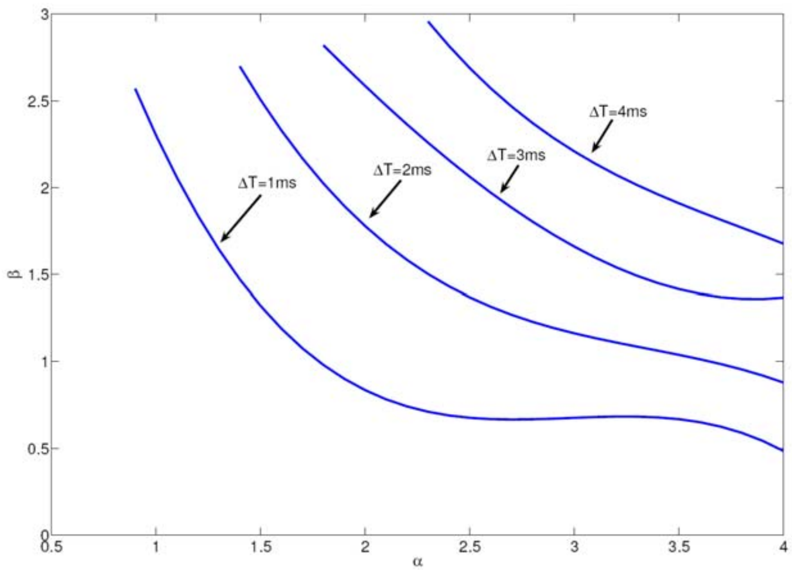


$\alpha = 2, \beta = 4, n = 100, \beta_2 = \beta + 1$

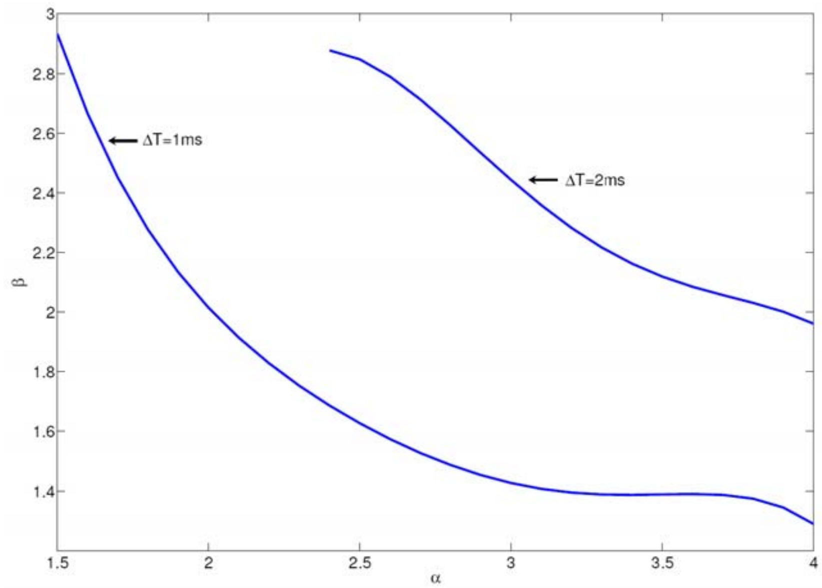


$\alpha = 2, \beta = 4, \Delta T = 0.1ms, \beta_2 = \beta + 1$

PUD Acceptable Performance Region



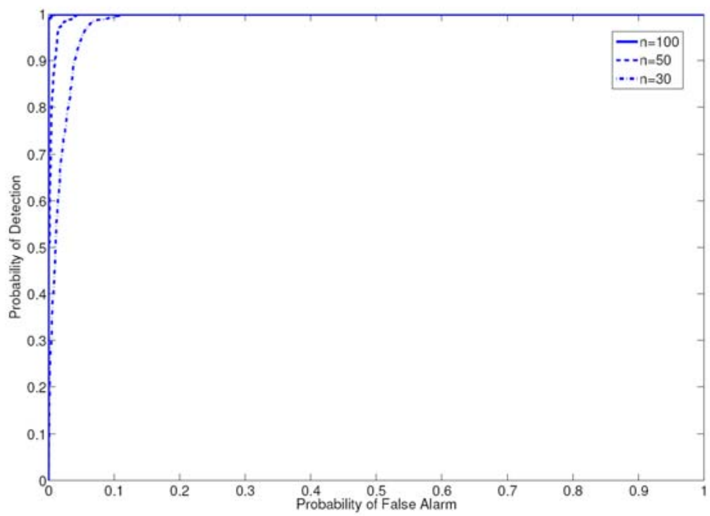
$n = 100$



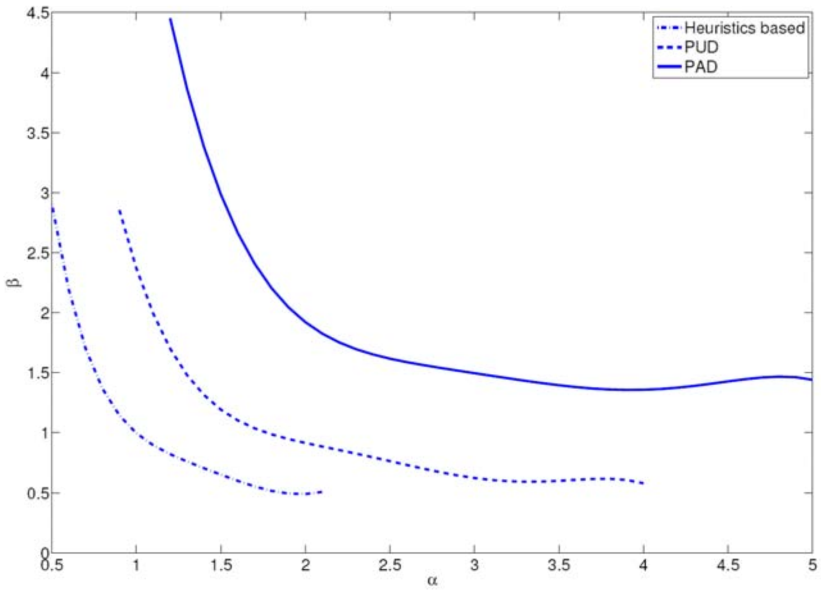
$n = 300$

Evaluation Using Internet Measurements

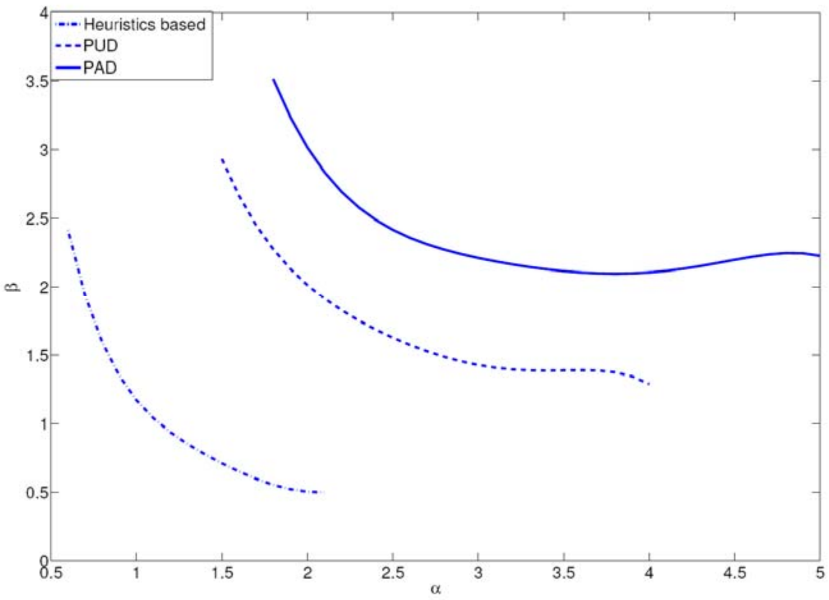
- Methodology
  - Collect RTTs using PlanetLab
  - Extract statistically homogeneous data
  - Segment into n sample windows
  - Calculate $L|H_0$ for each window
  - Add ΔT to last $\lceil n/2 \rceil$ samples
  - Calculate $L|H_1$ for each window
- Data
  - Athens, Greece – Tokyo, Japan
  - October 25, 2006
  - $\Delta T = 1\text{ms}$
  - n=100, acceptable performance



Acceptable Performance Region: All Three Algorithms



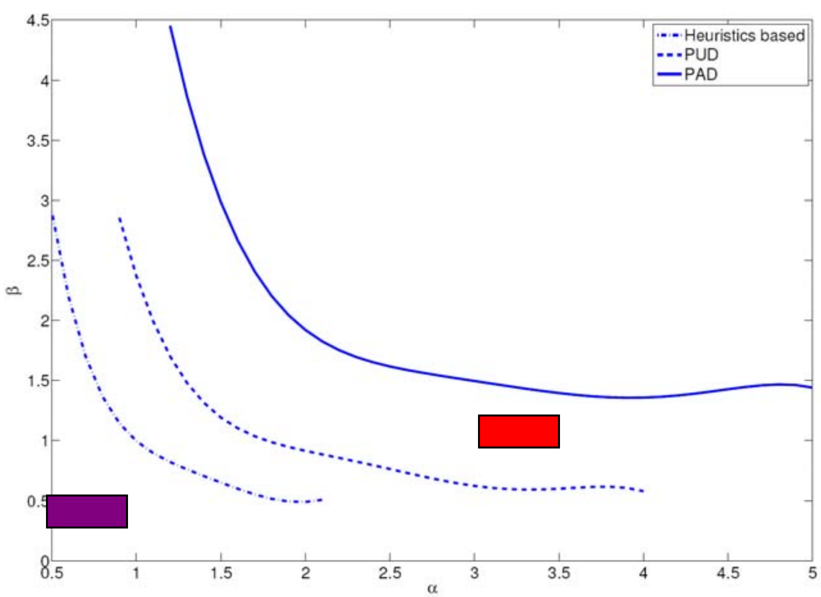
$n = 100, \Delta T = 1\text{ms}$



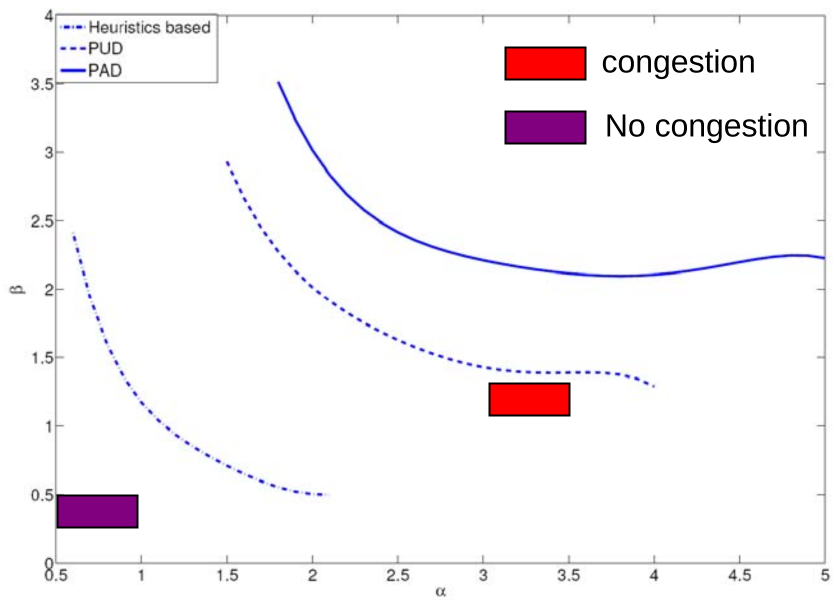
$n = 300, \Delta T = 1\text{ms}$

Minimum Sampling Rate

- Detect route changes 1 min apart
- Probing rate:
  - Congestion => 5 samples/sec
  - No congestion => 1.66 samples/sec
- Automation: estimate parameters – determine rate



$n = 100, \Delta T = 1\text{ms}$



$n = 300, \Delta T = 1\text{ms}$

Conclusions: Model-Based Approach

- Proposed optimal detector (PAD)
  - Developed analysis to predict performance
- Proposed practical detector (PUD)
- Performance evaluation
  - PUD performance region larger than heuristic
  - PUD performance region increases with n and approaches that of PAD
  - Heuristic performance region not sensitive to window size n

Conclusions

- Developed methods to detect impairments
  - Evaluated using PlanetLab data
- Proposed/evaluated heuristic methods
  - Congestion and Route changes
- Proposed optimal detector (PAD)
  - Developed analysis to predict performance
- Proposed practical detector (PUD)
  - Performance better than heuristics-based
- Discovered the impairment vulnerability of PF
  - Proposed parallel PF and adaptive alpha initialization

Future Work

- Analysis to predict performance of PUD
  - Analysis is difficult
  - Make some simplifying assumptions
  - Useful in predicting threshold in real-time
- PAD to detect changes during route flaps
  - When to start applying PAD
  - Detect when flapping stops and apply PUD

Publications

- Soshant Bali, Yasong Jin, Victor S. Frost, Tyrone Duncan, “Characterizing User-Perceived Impairment Events Using End-to-End Measurements,” International Journal for Communication Systems, Vol. 18, No. 10, pp. 935-960, December 2005.
- Soshant Bali, Sridhar Machiraju, Hui Zang and Victor Frost, “A Measurement Study of Scheduler-Based Attacks in 3G Wireless Networks,” Passive and Active Measurements Conference, Louvain-La-Neuve, Belgium, May 5-6, 2007.
- Soshant Bali, Sridhar Machiraju and Hui Zang, “Beyond Proportional Fair: Designing Robust Wireless Schedulers,” poster in IFIP Networking, 2007.
- S. Bali, S. Machiraju and H. Zang, "PAQ: A Starvation Resistant Alternative to Proportional Fair," Submitted to ICC 2008.
- Soshant Bali and Victor S. Frost, “Model-based detection of Route Changes,” submitted to IEEE/ACM Transactions on Networking.
- Soshant Bali and Victor S. Frost, “A New Algorithm for Fitting MMPP to IP Traffic Traces,” IEEE Communications Letters, Vol. 11, No. 2, pp. 207-209, Feb. 2007.
- Yasong Jin, Soshant Bali, Tyrone Duncan and Victor Frost, “Predicting Properties of Congestion Events for a Queuing System with fBm Traffic,” IEEE/ACM Transactions on Networking, Vol. 15, No. 5, pp. 1098-1108, October 2007.