NF-TCP: Network Friendly TCP

Mayutan Arumaithurai
Xiaoming Fu
K. K. Ramakrishnan

Motivation I

- P2P file sharing applications
  - Open multiple connections
  - Download large volume of data
  - **Delay insensitive** (download and play)
- TCP: fairness for all flows
  - Does not distinguish between types of flows (Both from technological and user's point of view)
    - Delay-sensitive applications (e.g. video, voice, web traffic)
    - Delay-insensitive applications (P2P, backup, software updates)Users resort to manual means

Motivation II

- Currently, there is no means for a delay-sensitive (P2P) application to be submissive to delay-sensitive applications during congestion periods
- Therefore attempts to reduce the effect of P2P traffic include
  - Users
    - Manually switch on and off
  - Service providers
    - Block P2P traffic
    - Throttle P2P traffic

Design goals/requirements

- Be submissive to Standard TCP flows
- Detect incipient congestion
- Use available bandwidth when network not congested
- Be fair to other NF-TCP flows
- Adaptive

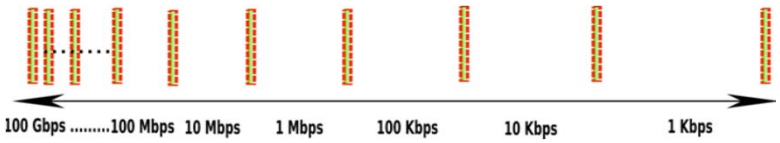
Related Work

- Protocols that are designed for Delay-insensitive applications
  - LEDBAT, TCP-LP
- Approaches that use Bandwidth quickly in High BDP networks
  - Aggressive increase: Highspeed TCP, CUBIC TCP, Compound TCP
  - Router Based: Quickstart TCP, VCP, BMCC
  - Probing based: RAPID
- Note: All the above need to cause congestion to identify target load that could be placed

Network Friendly TCP (NF-TCP)

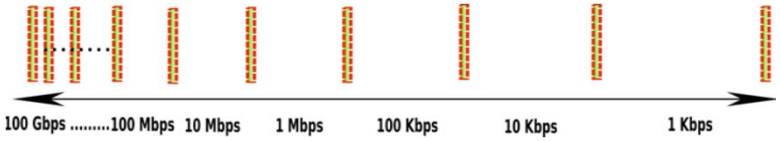
- Measurement based Bandwidth Estimation
  - Identify available bandwidth without causing congestion
- ECN marking based congestion detection
  - To provide early indication
- Flow control
  - Non-congestion periods: Depends on Bandwidth estimate instead of only on current load
  - Congestion periods: Similar to standard TCP
- Submissive Mode
  - Combination of early detection and TCP like reduction of congestion window
  - Because we respond early to congestion

NF-TCP Bandwidth estimation - I



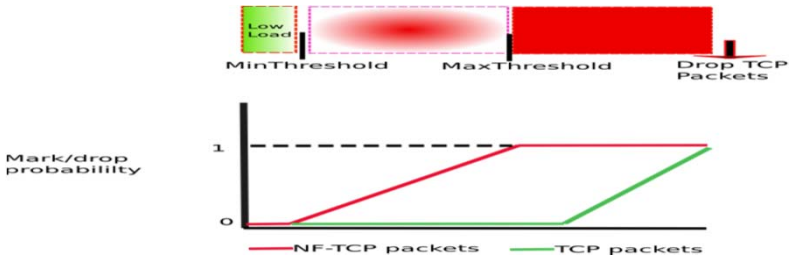
- Measurement based bandwidth estimation
  - To guide the amount of load placed on the network
- Not depend on the P2P application to cause congestion to detect available bandwidth

NF-TCP Bandwidth estimation - II



- Probing mechanism
  - Pathchirp/RAPID like mechanism
  - Sending: Decrease in inter-packet delay between packet pairs
  - Receiver
    - If inter-packet delay decreases => BW_est >= Rate(k)
    - If inter-packet delay increases (queuing) => BW_est <= Rate(k)
  - Dynamic probing (?) => Next probe's Minrate = beta * W

NF-TCP Congestion Detection

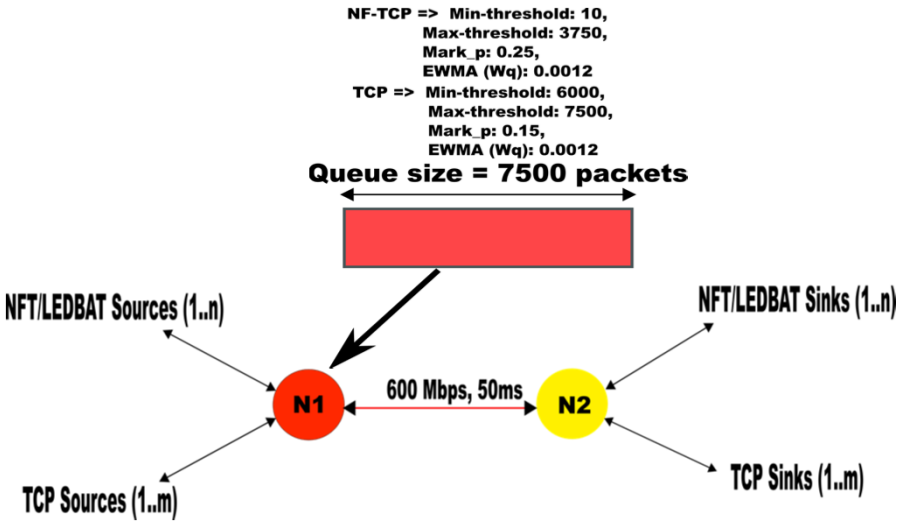


- Goal
  - Standard TCP shouldn't be impacted
  - Identify incipient congestion
  - Signal it to the NF-TCP sender
- RED parameters Minthresh, MaxThresh, mark_prob changed
- Timescale (still to be studied)
- EWMA for NF-TCP different from TCP (still to be studied)

NF-TCP flow control

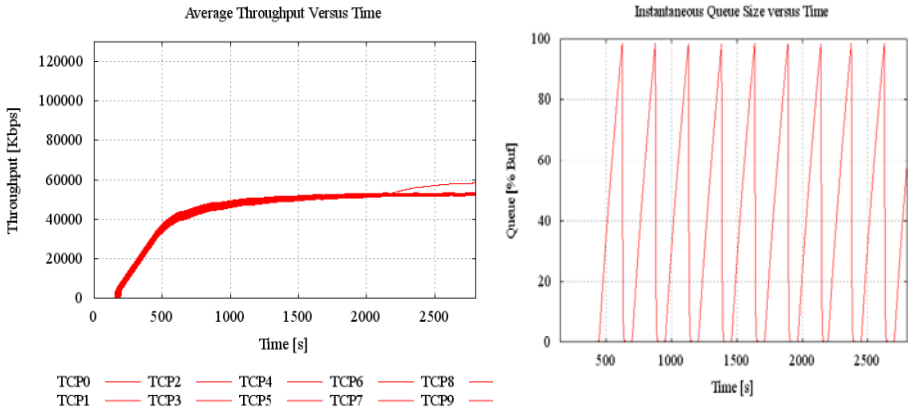
- Use BW_est to perform aggressive increase to opportunistically use bandwidth
  - Non-congestion periods: w= w + alpha*BW_est*RTT
- Quickly respond to congestion and be submissive
  - When congested: w = w – 0.5 * w (We haven't changed this behaviour for NF-TCP to reduce oscillations when only NF-TCP is present)
- => Use bandwidth opportunistically when available, and backoff quickly on detecting congestion

Simulation Topology



Lot more evaluations being performed

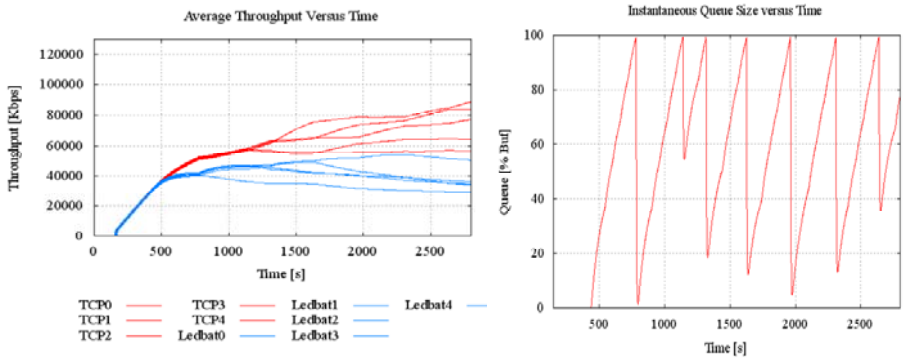
ALL-TCP (10 flows)



All flows share the bandwidth equally

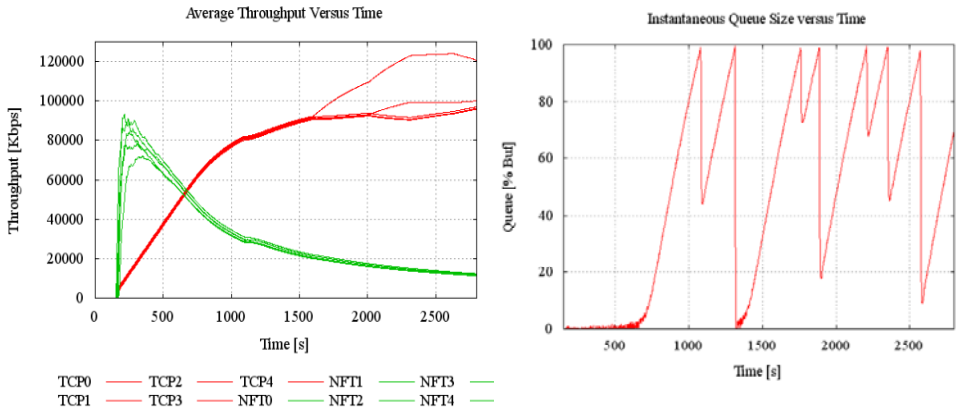
LEDBAT (5 flows) vs TCP (5 flows)

- Uses Delay estimation for determining congestion
- Currently work in progress at the IETF Ledbat WG
- Intended to back off earlier than standard TCP on noticing increase in delay



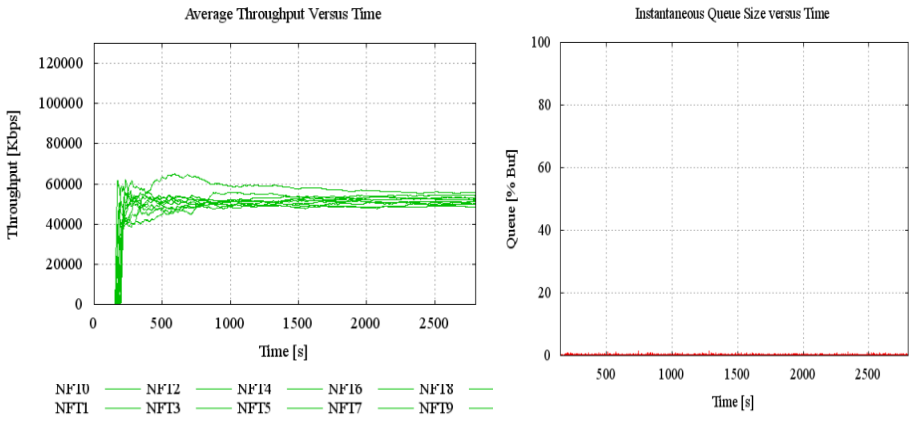
Backs off, but introduces variability for TCP too

NF-TCP (5 flows) vs TCP (5 flows)



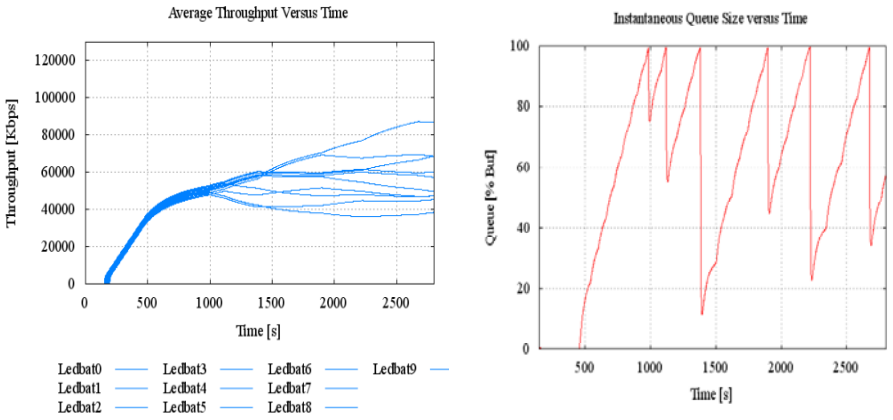
NF-TCP is submissive to Standard TCP and TCP throughput is much higher

All NF-TCP (10 flows)



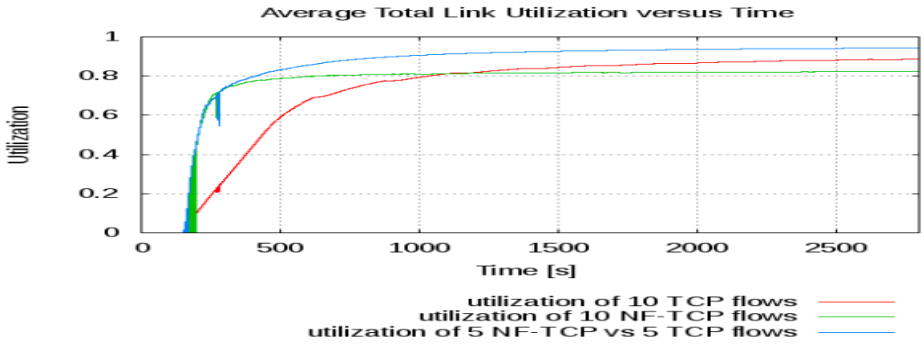
All flows share the bandwidth equally and the queue is maintained very close to zero

All LEDBAT (10 flows)



There is some variability

Utilization



NF-TCP coexisting with TCP is able to opportunistically use bandwidth and increase utilization

In the case of only NF-TCP flows, the utilization is low since the queues are maintained at near zero

Summary

- Studied if measurement can guide congestion control
- Application: Used it to develop NF-TCP for delay-insensitive applications
- Further work
  - Lots to do

Thank you

Comparison to RAPID

- RAPID is completely probe/rate based
  - NF-TCP uses probing to assist in identifying available bandwidth, but not to detect congestion
- RAPID is not designed to be entirely submissive to standard TCP
- RAPID detects congestion with the help of a delay based probing mechanism
  - NF-TCP detects congestion with the help of an ECN enabled RED queue.