

Enhancing TCP Fairness in Ad Hoc Wireless Networks Using Neighborhood RED

Kaixin Xu, Mario Gerla
University of California, Los Angeles
{xkx, gerla}@cs.ucla.edu

Lantao Qi, Yantai Shu
Tianjin University, Tianjin, China
{ltqi, ytshu}@tju.edu.cn

***This work was supported in part by Office of Naval Research (ONR) "MINUTEMAN" project under contract N00014-01-C-0016 and TRW under a Graduate Student Fellowship**

***This research was supported in part by the National Natural Science Foundation of China (NSFC) under grant No. 90104015.**

Motivation

- TCP is important to ad hoc networks
 - Reliable transfer of data/image files and multimedia streaming
 - Congestion protection
 - Efficient utilization and fair share of the resources
- Significant TCP unfairness has been reported in Last several years

TCP Unfairness in Ad Hoc Networks

- Fairness index in wireless networks

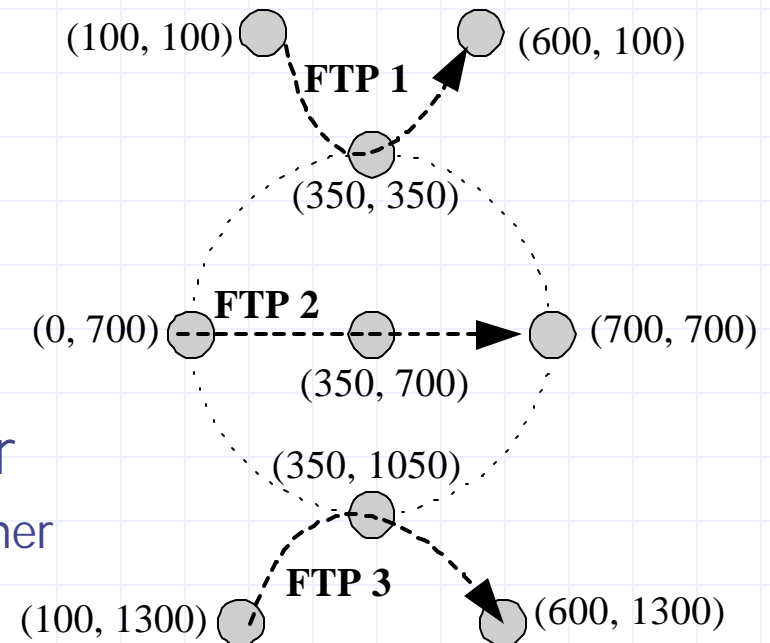
- Weighted MaxMin Fairness Index

- Weight(i) = # of contending flows of flow i (including itself)

- $$F(X,t) = \frac{\left[\sum_{i=1}^n w_i(t) X_i(t) \right]^2}{n \left[\sum_{i=1}^n (w_i(t) X_i(t))^2 \right]}$$

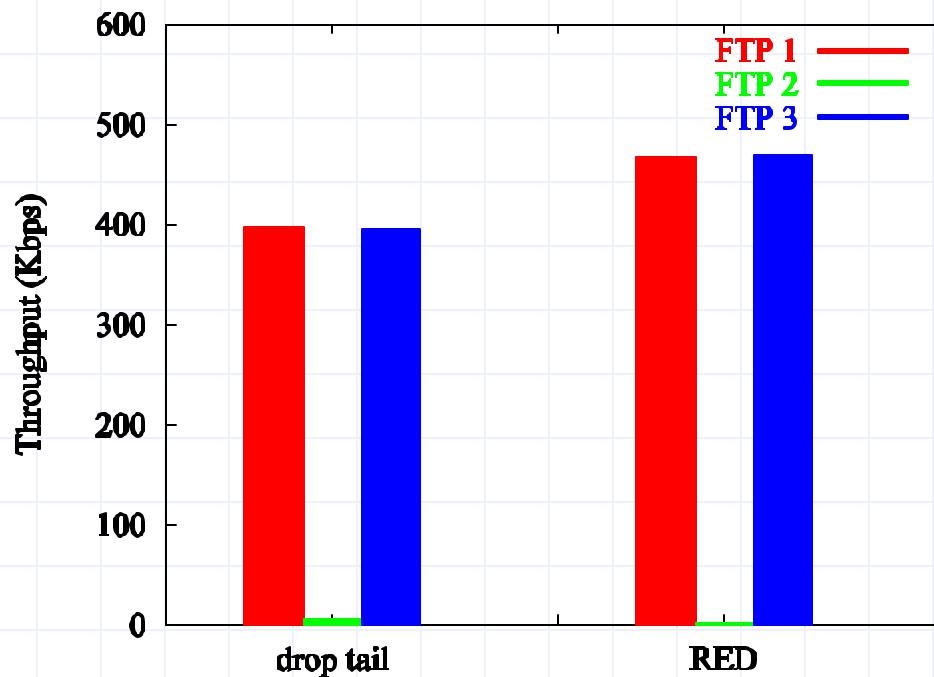
- Simulation in QualNet simulator

- 3 TCP flows contending with each other
 - Weight of 3 flows, 2:3:2



Significant TCP Unfairness

- Flow 2 is nearly starved
- Original RED fails to improve the fairness
- Weighted Fairness Index = 0.67

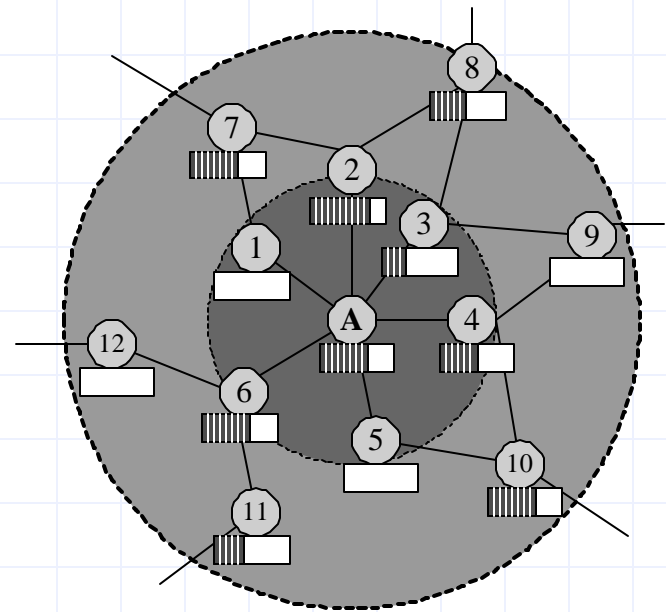


Why RED Does Not Work?

- Random Early Detection (RED)
 - Active queue management scheme
 - Average queue size: $avg = (1 - w_q) * avg + w_q * q$
 - Drop probability: $p_b = \frac{\max_p(avg - \min_{th})}{\max_{th} - \min_{th}}$, proportional to buffer occupancy
- Why RED does not work in ad hoc networks?
 - Congestion simultaneously affects multiple queues
 - Queue at a single node cannot completely reflect the state
- Extend RED to the entire congested area – Neighborhood of the node

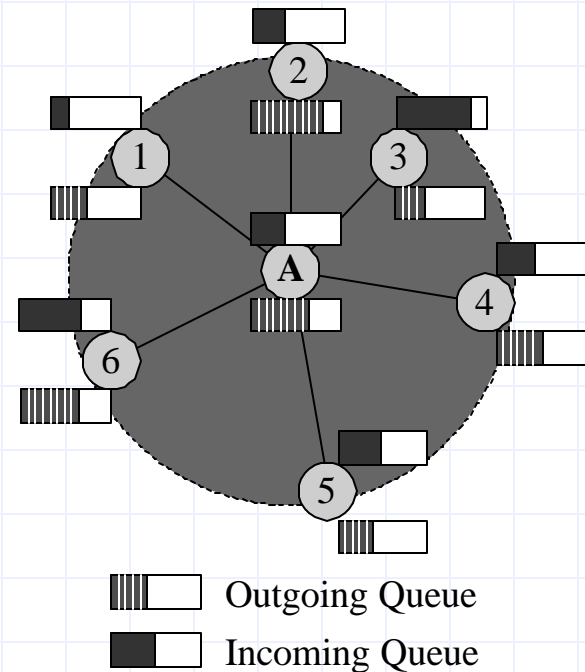
Neighborhood and Its Distributed Queue

- A node's neighborhood consists of the node itself and the nodes which can interfere with this node's signal
 - 1-hop neighbors directly interfere
 - 2-hop neighbors may interfere
- Queue size of the neighborhood reflects the degree of local network congestion



Simplified Neighborhood Queue Model

- 2-hop neighborhood queue model is not easy to operate
 - Too much overhead
 - Only some packets in 2-hop neighbors' queues should be counted
- Simplified model
 - Only include 1-hop neighbors
 - Two queues at each neighbor
- Distributed neighborhood queue – the aggregate of these local queues



Characteristics of Neighborhood Queue

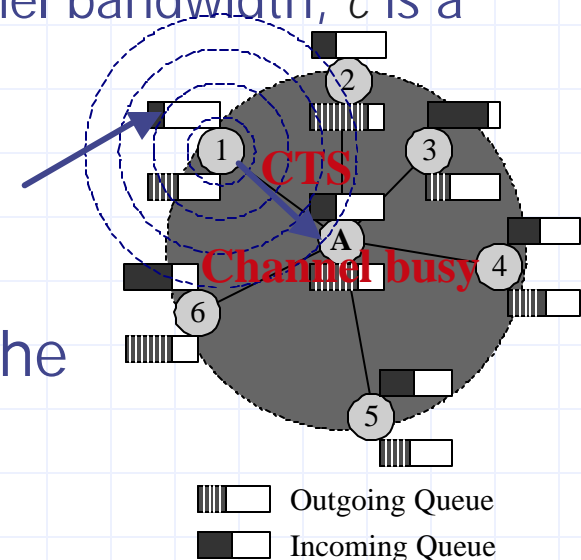
- Consists of multiple queues located at the neighboring nodes
- Not a FIFO queue due to location dependency
- Priority of a sub-queue may change dynamically
 - Topology changes
 - Traffic pattern changes
- TCP flows sharing the same neighborhood may get different feedbacks in terms of packet delay and loss rate

Neighborhood Random Early Detection (NRED)

- Extending RED to the distributed neighborhood queue
- Key Problems
 - Counting the size of the distributed neighborhood queue
 - Calculating proper packet drop probability at each node
- Components of Neighborhood RED
 - Neighborhood Congestion Detection (NCD)
 - Neighborhood Congestion Notification (NCN)
 - Distributed Neighborhood Packet Drop (DNPD)

Neighborhood Congestion Detection

- Direct way: Announce queue size upon changes
 - Too much overhead, exacerbates congestion
- Our method: Indirectly estimate an index of queue size by monitoring wireless channel
 - Channel utilization ratio $U_{busy} = \frac{\text{channel} - \text{busy} - \text{time}}{\text{sampling} - \text{interval}}$
 - Queue size index $q = \frac{U_{busy} * W}{c}$, W is channel bandwidth, c is a constant packet size
- Average queue size is calculated using RED's alg.
- Congestion: queue size exceeds the minimal threshold

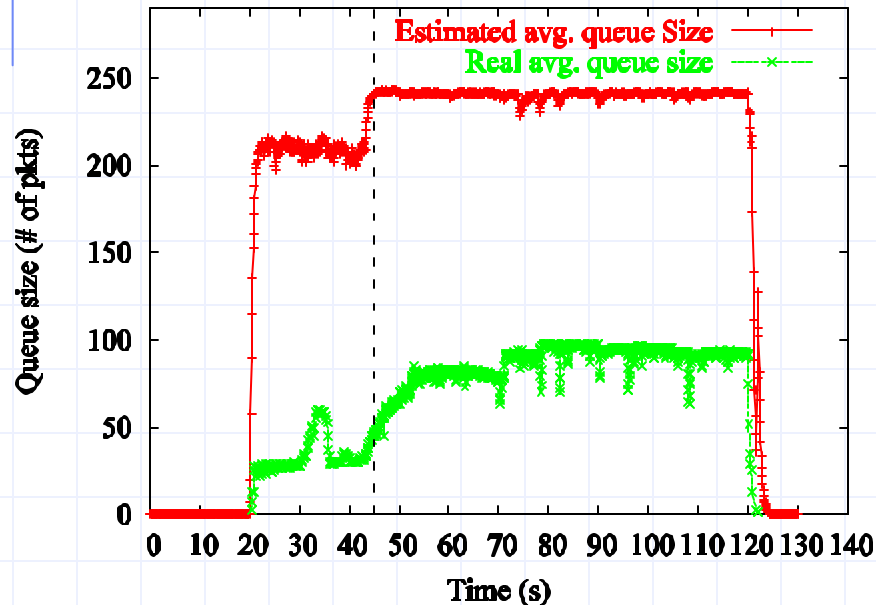
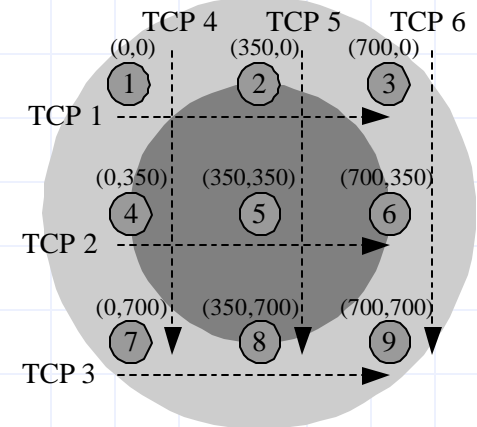


Neighborhood Congestion Notification & Distributed Neighborhood Packet Drop

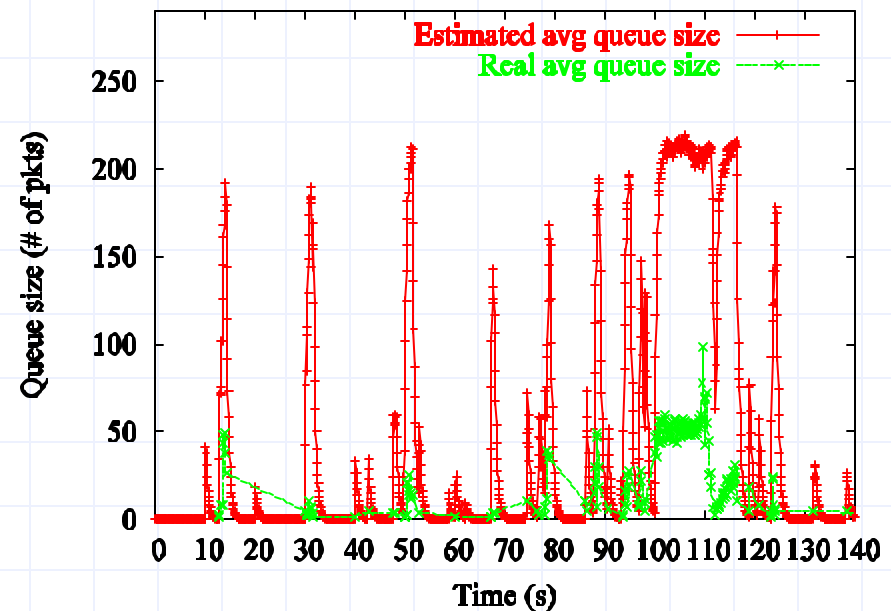
- Neighborhood Congestion Notification
 - Drop probability over the whole neighborhood queue following RED's alg.
 - Broadcast the drop probability to 1-hop neighborhoods
- Distributed Neighborhood Packet Drop
 - Local drop probability is proportional to local node's channel usage

Verification of Queue Size Estimation

- Estimating Node5's neighborhood queue size index
- Get real queue size by recording queue size at all nodes



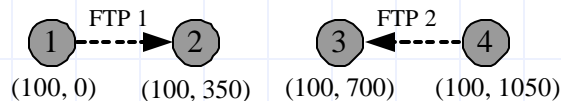
FTP Traffic



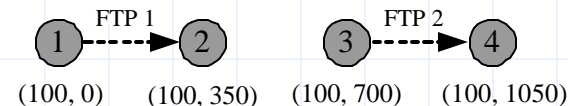
HTTP Traffic

Parameter Tuning: Scenarios

- QualNet simulator
- Basic but typical scenarios
 - Hidden terminal situations
 - Expose terminal situations
- Configuration parameters
 - Minimum threshold & Maximum threshold
 - Set to 100 and 240 based on previous experiment
 - Vary the maximum packet drop probability (\max_p)



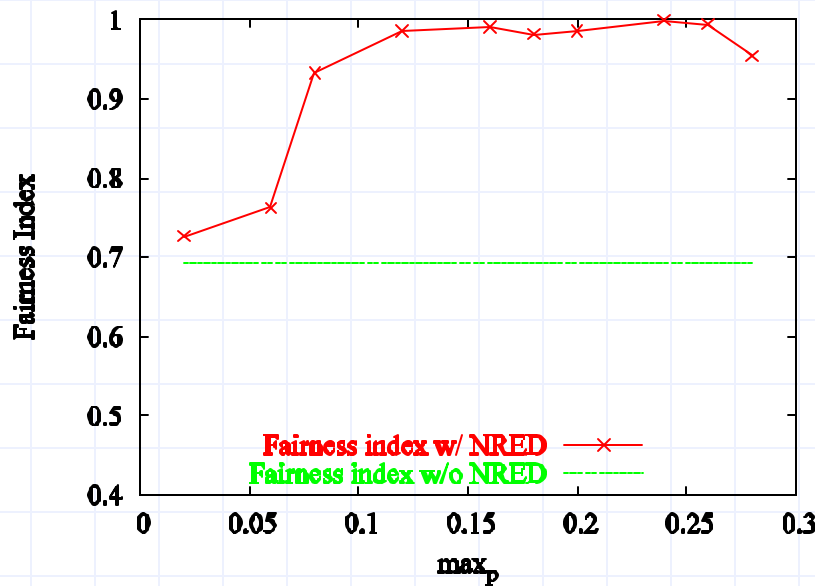
Hidden Terminal



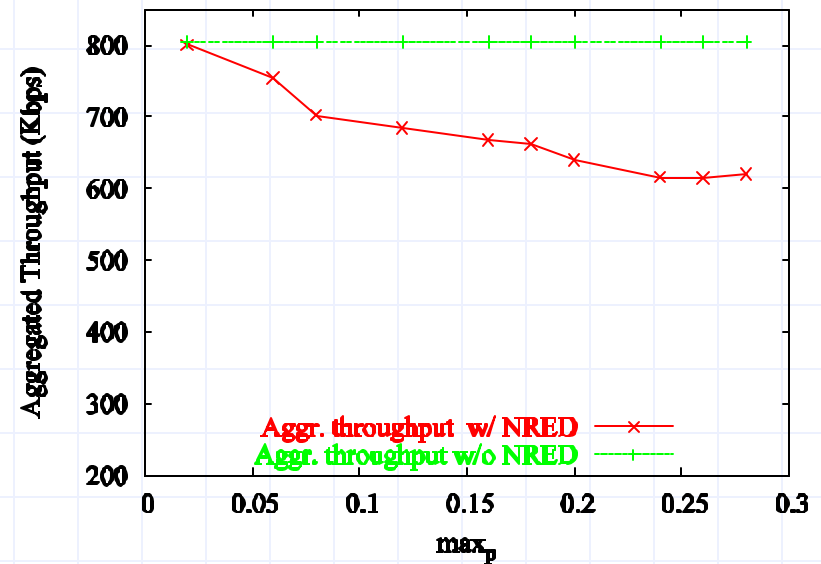
Exposed Terminal

Parameter Tuning: Hidden Terminal Scenario

- Weighted fairness index
- Instantaneous throughput: $X(t) = \frac{D_t}{\Delta_t}$, here Δ_t denotes the data successfully received during time period $[t \rightarrow t + \Delta_t]$

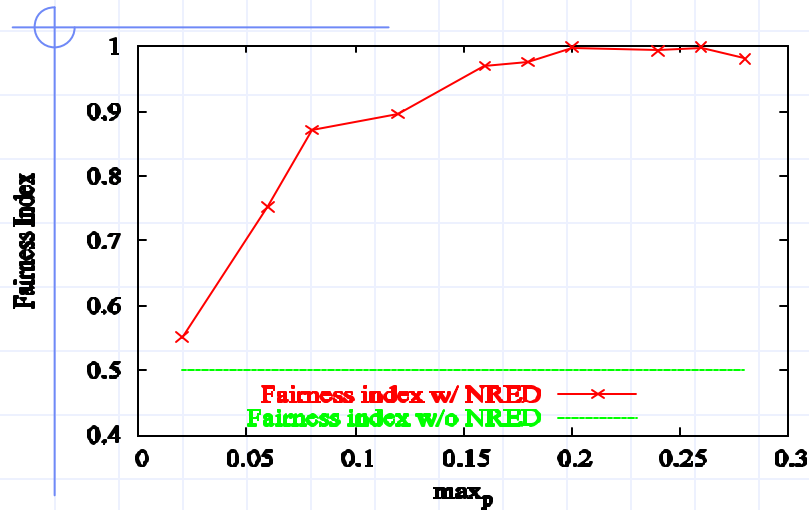


Fairness index

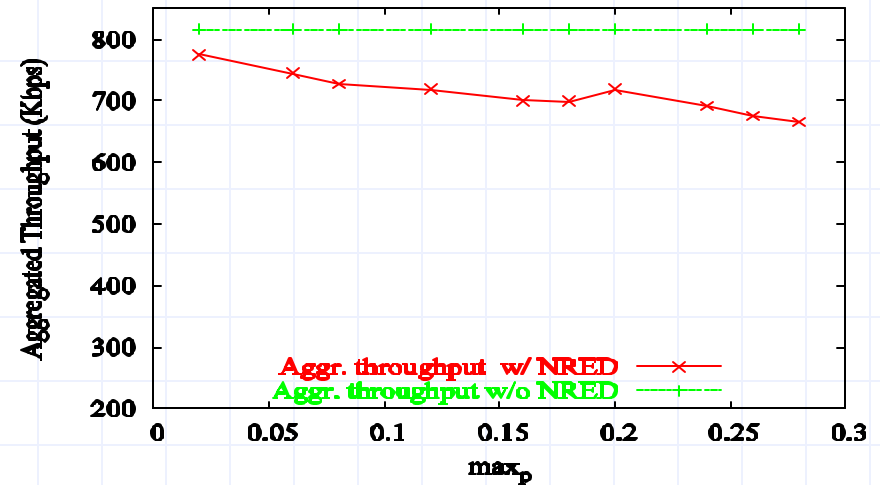


Aggregated throughput

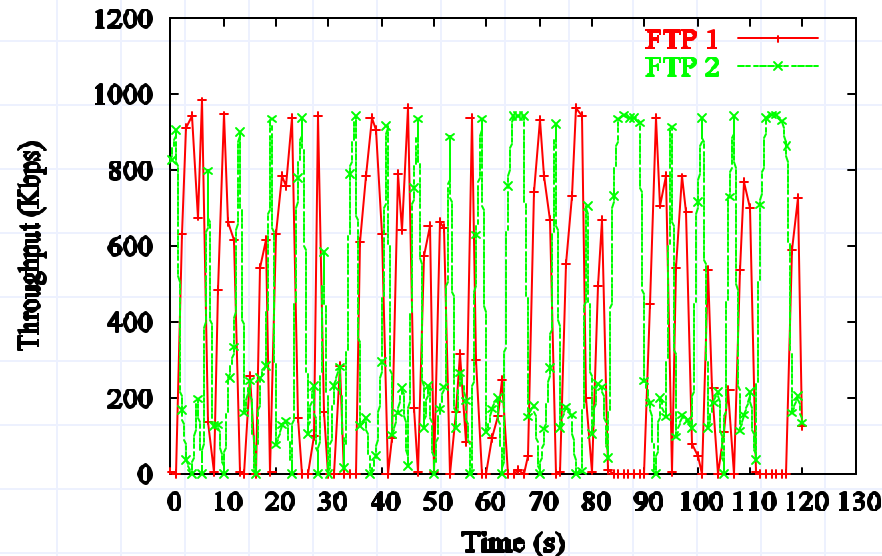
Parameter Tuning: Exposed Terminal Scenario



Fairness index



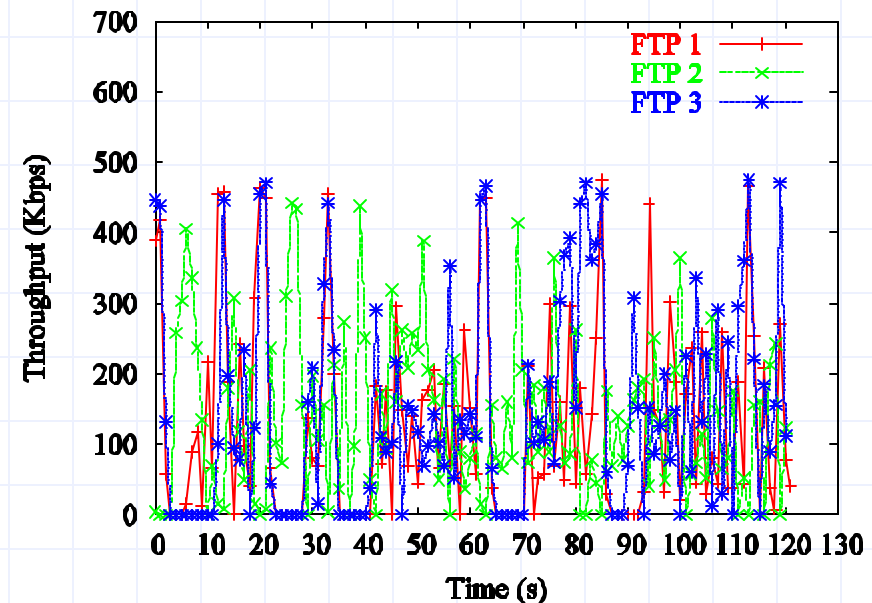
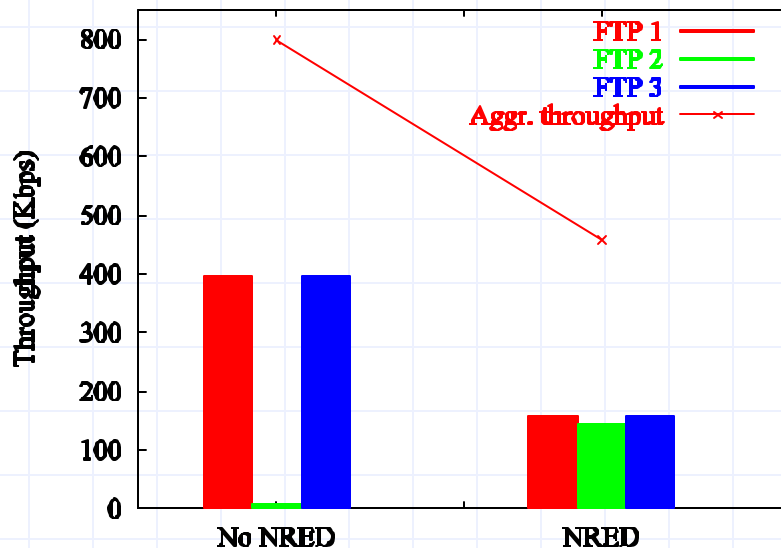
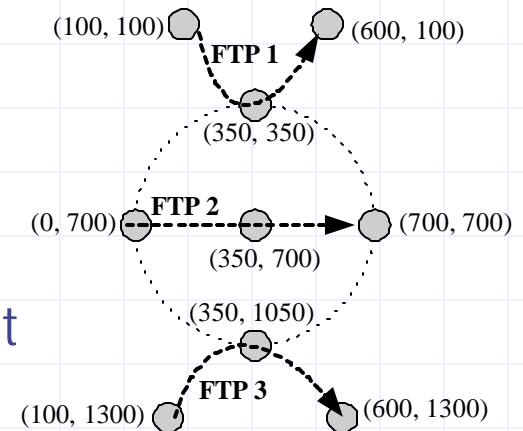
Aggregated throughput



Instant throughput
W/ $\max_p = 0.14$

Performance Evaluation: Simple Scenario

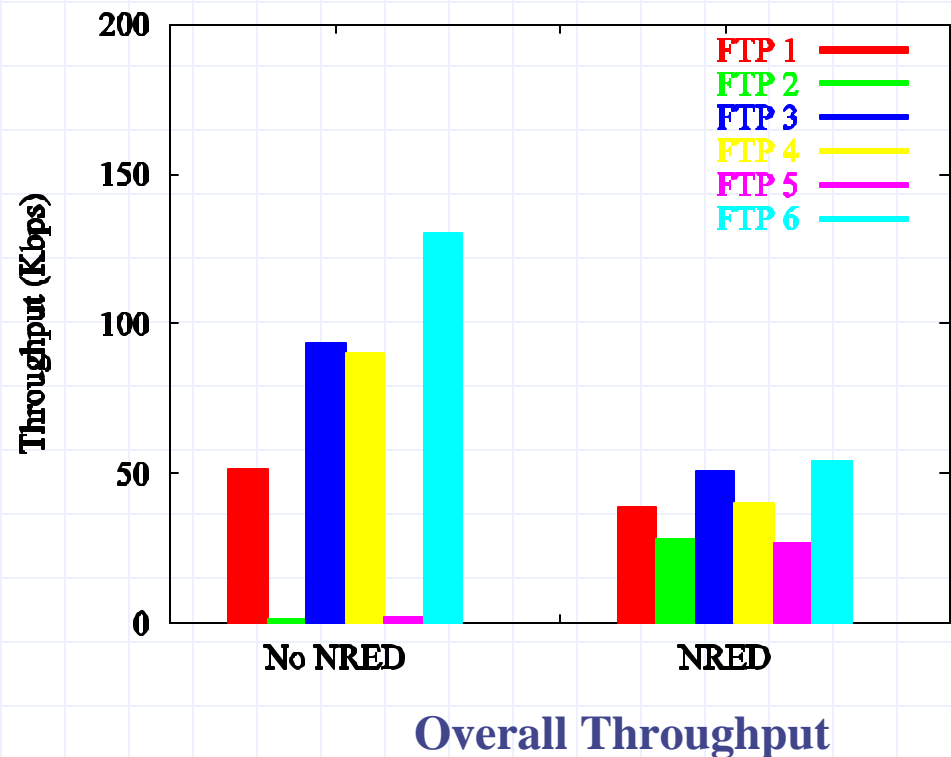
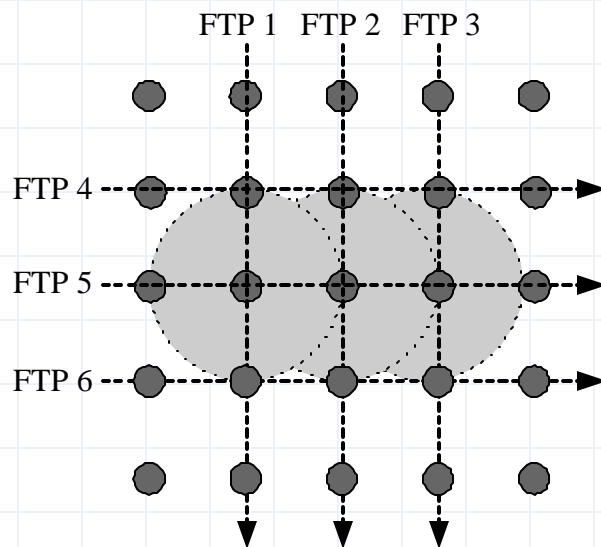
- Both long-term and short-term fairness is achieved
- Loss of aggregated throughput
 - Tradeoff between fairness and throughput
 - Channel is not fully utilized



Instantaneous Throughput 16

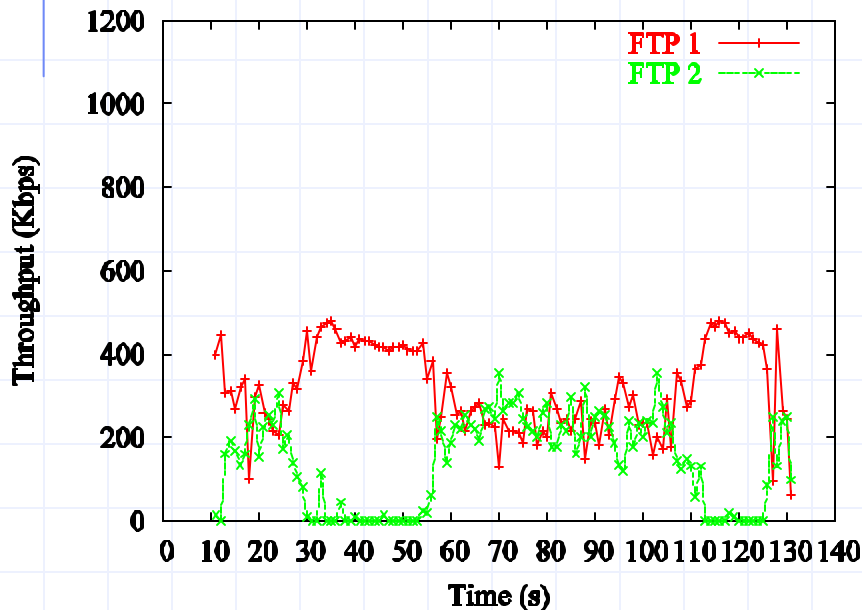
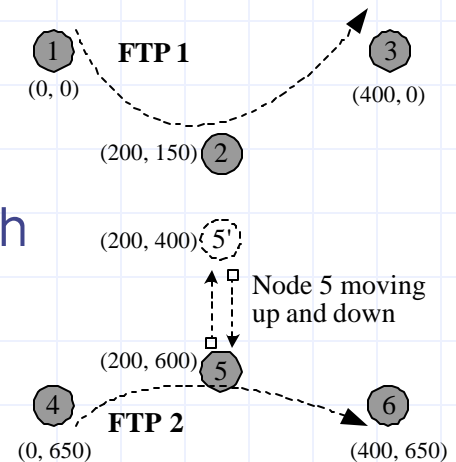
Performance Evaluation: Multiple Congested Neighborhood

- Multiple congested neighborhoods
- FTP2 & FTP 5 have more competing flows, are more likely to be starved

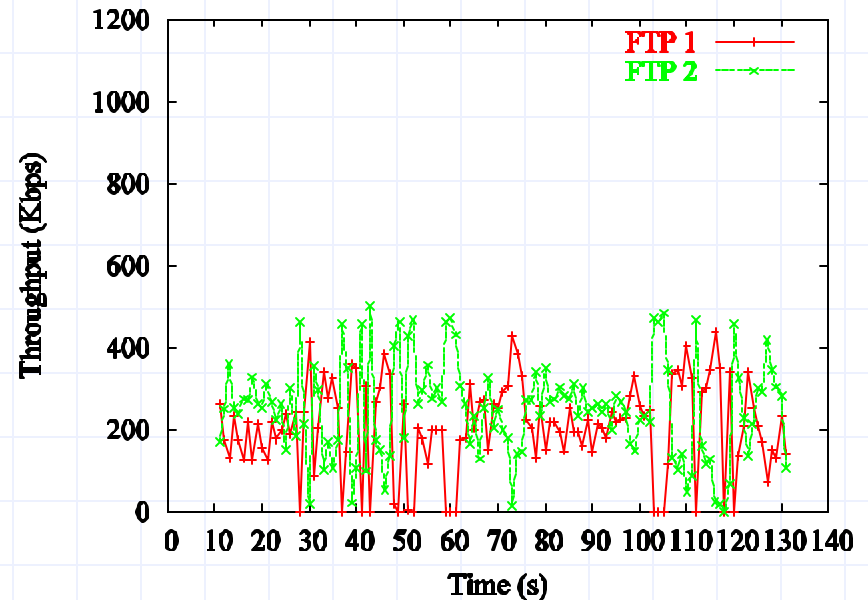


Performance Evaluation: Mobility

- Node 5 moves up and down
 - Moving Up: two flow interfere with each
 - Moving down: No much interference
- NRED can adapt to mobility



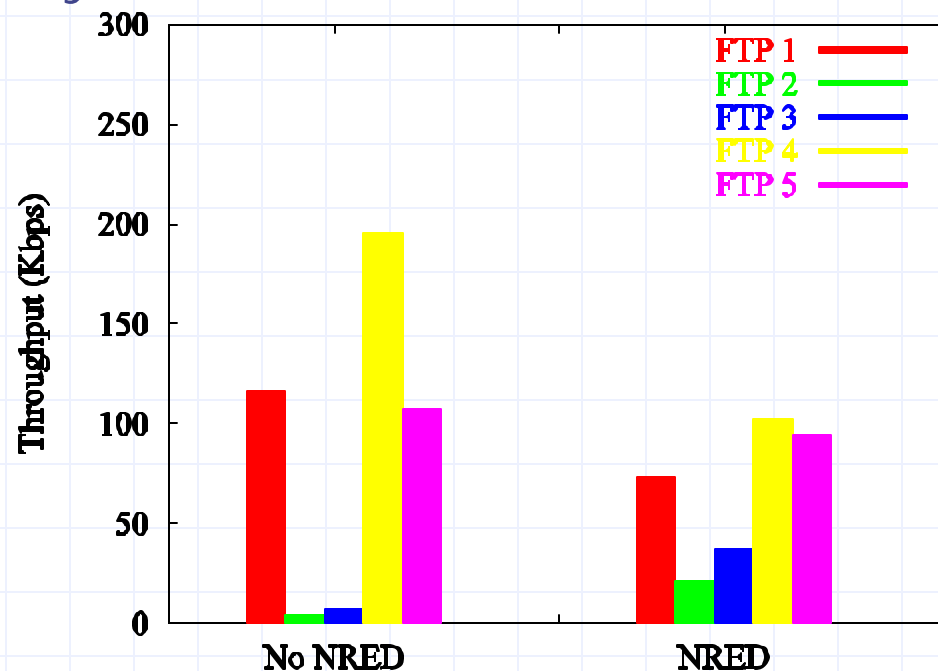
Without NRED



With NRED

Performance Evaluation: Realistic Scenario

- 50 nodes randomly deployed in 1000mX1000m field
- 5 FTP/TCP connections are randomly selected
- AODV routing
- No mobility



Conclusions

- Significant TCP unfairness has been found and reported in ad hoc networks
- NRED is a network layer solution
 - Easy to implement
 - Incremental deployment
- Major Contributions
 - Model of neighborhood queue
 - Distributed neighborhood queue
 - Not FIFO, different and dynamic priorities
 - Network layer solution for enhancing TCP fairness in ad hoc networks



Thanks!