LCA

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

Enhance & Explore: an Adaptive Algorithm to Maximize the Utility of Wireless Networks

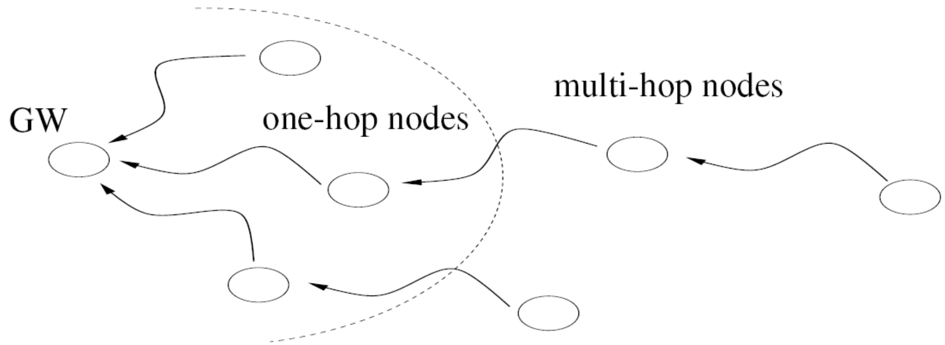
Adel Aziz

joint work with
Julien Herzen, Ruben Merz, Seva Shneer, and Patrick Thiran

September 21st 2011

CH-1015 Lausanne
adel.aziz@epfl.ch
http://people.epfl.ch/adel.aziz

Problem Statement



$$\boxed{U_{max}(\vec{x}[n]) = \sum_{i=1}^{F} x_i[n]}$$

$$\boxed{U_{prop}(\vec{x}[n]) = \sum_{i=1}^{F} \ln(x_i[n] + 1)}$$

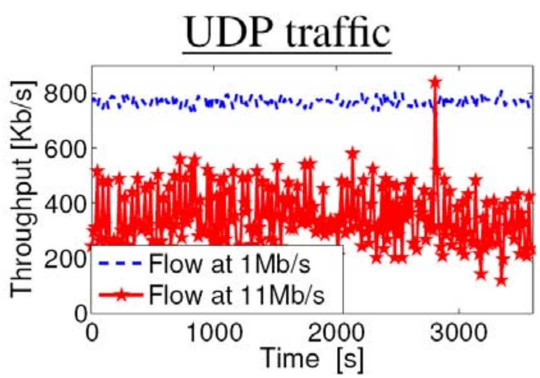
- What is the network utility?
  - Based on per-flow throughput $\vec{x}[n] \in \mathbb{R}_+^F$
  - Captures an objective

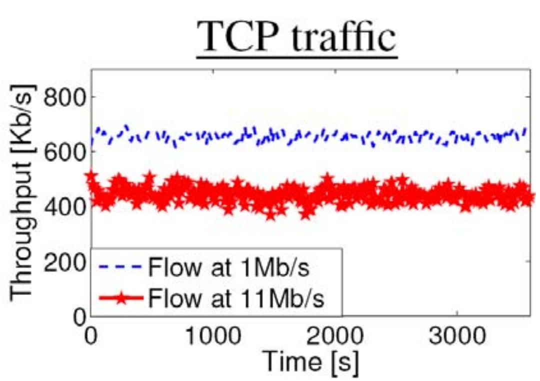
- How to maximize the network utility?
  - Avoid wasted resource (e.g. buffer overflow)
  - Optimal share of resource between flows

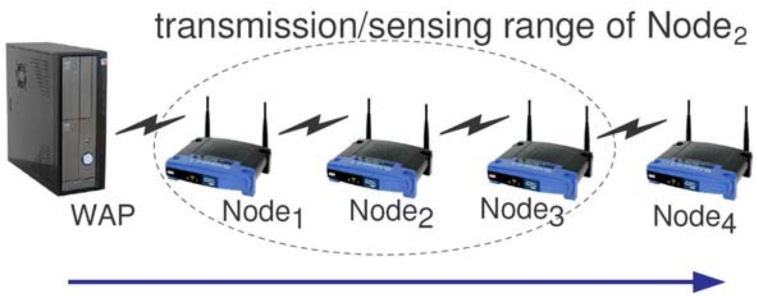
Problem Motivation

WLAN Setting

- Inter-Flow Problem

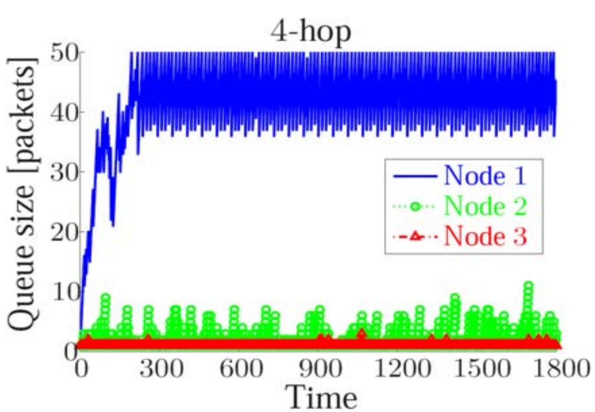






Multi-hop Setting

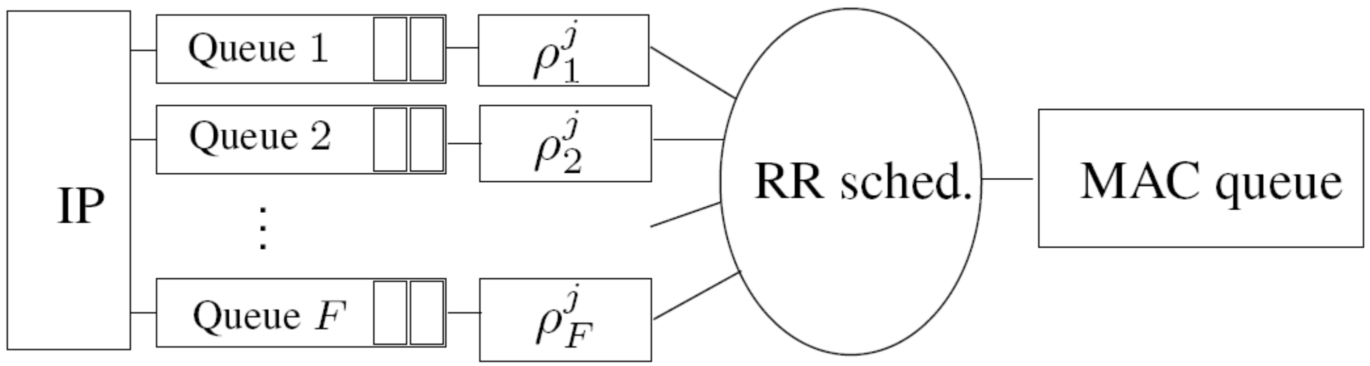
- Intra-Flow Problem



Network Model

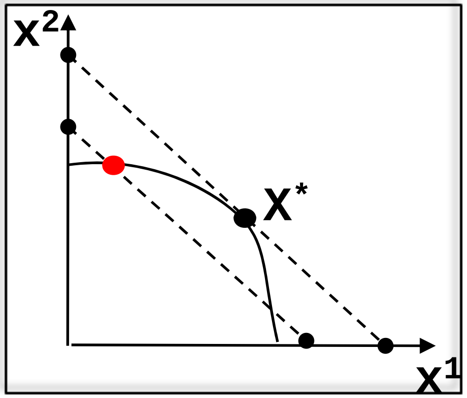
- Flow = <IP src; IP dst>

- Architecture of node *j*



- Information to set at node *j* at time slot *n*
  - Rate allocation vector: $\vec{\rho}[n] \in \mathbb{R}_+^F$
- Information to monitor at GW at time slot *n*
  - Measured throughput: $\vec{x}[n] \in \mathbb{R}_+^F$

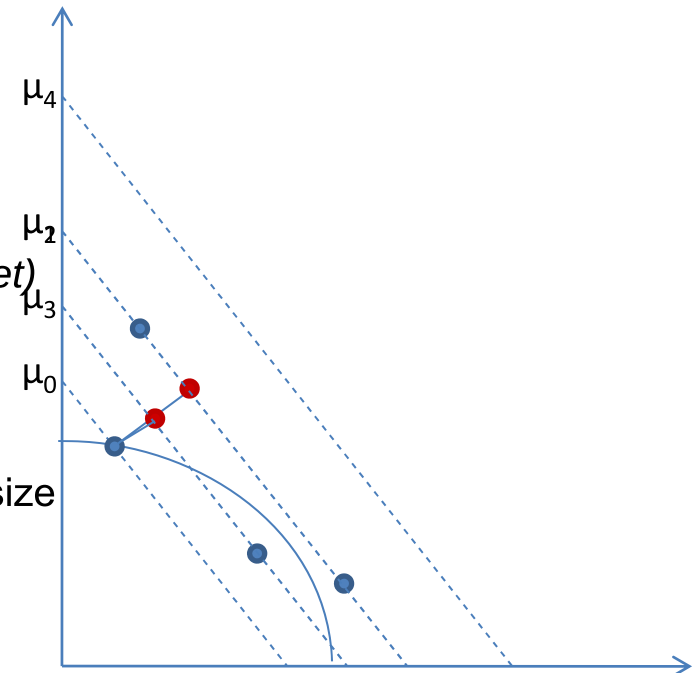
Network Model



- Useful information at time slot *n*
  - Measured throughput: $\vec{x}[n] \in \mathbb{R}_+^F$
  - Capacity region: $\Lambda[n]$ (unknown *a priori*)
  - Rate allocation vector: $\vec{\rho}[n] \in \mathbb{R}_+^F$
  - Last stable allocation: $\vec{r}[n] \in \mathbb{R}_+^F$
  - Utility function: $U_{max}(\vec{x}[n]) = \sum_{i=1}^{F} x_i[n]$
  - Level set: $L(\mu[n]) = \{\vec{x}[n] : \vec{x}[n] \in \mathbb{R}_+^F, U(\vec{x}[n]) = \mu[n]\}$

‘Enhance & Explore’ in WLAN

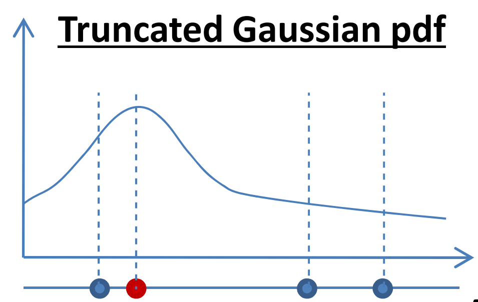
- ❑ Starts from IEEE 802.11 allocation
  - o Throughput vector: $x_0 = \rho_0$ [rate allocation]
  - o Level set of utility μ_0: $L(x_0, \mu_0)$ [$u(x_0) = \mu_0$]
  - o Remember allocation: $r_0 = x_0$
  - o Move to next slot: 0 -> 1



- ❑ Enhance phase: *(Find the next targeted level set)*
  - o If ($x_{i-1} = \rho_{i-1}$)
    - ▪ μ_i obtained by a full-size gradient ascent
  - o Else
    - ▪ μ_i obtained by dividing by 2 the last gradient ascent size
  - o Go to Explore phase

- ❑ Explore phase: *(Find the next allocation)*
  - o Pick point ρ_i randomly in $L(\rho_i, \mu_i)$
  - o Measure throughput: x_i
  - o If ($x_i = \rho_i$)
    - ▪ Remember new allocation: $r_i = x_i$
    - ▪ Go to Enhance phase and move to next slot (i -> i+1)
  - o Else
    - ▪ Move to next slot: i -> i+1
    - ▪ Keep old variables: $\mu_i = \mu_{i-1}$; $r_i = r_{i-1}$;
    - ▪ Repeat Explore phase at most **N** times, then move to Enhance

Example: **N** = 2



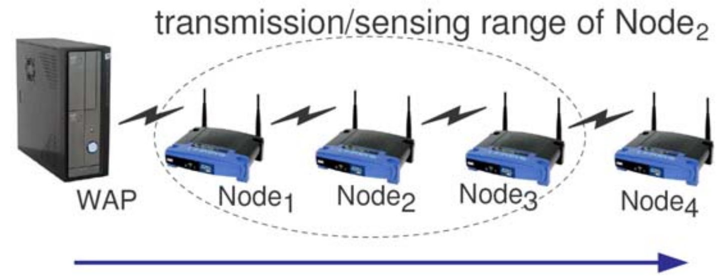
Optimality Theorem

- Theorem

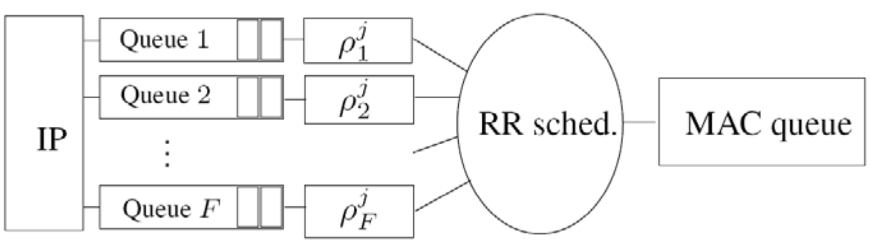
> THEOREM 3.1. *Assume that the capacity region Λ is fixed and coordinate-convex. Then the E&E algorithm guarantees that, for any initial rate allocation $\vec{r}[0]$, the utility of the last stable allocation $\vec{r}[n]$ converges to the maximal utility for $n \to \infty$.*

- Assumptions
  - Fixed capacity region $\Lambda[n] = \Lambda$
  - Coordinate-convex capacity region
    - Much weaker than convexity!

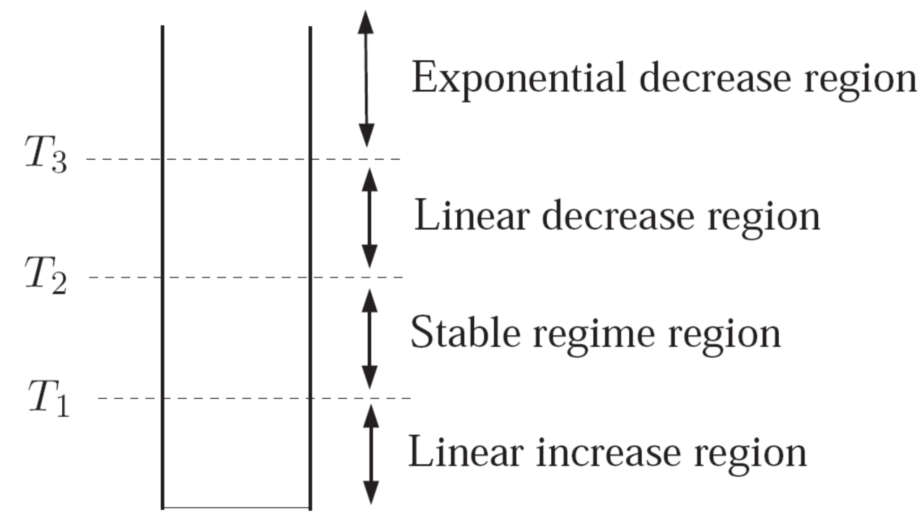
EZ-Flow in Multi-Hop



- Nodes passively monitor the next-hop queue[1]
- Nodes adapt their rate limiter ρ_i^j accordingly

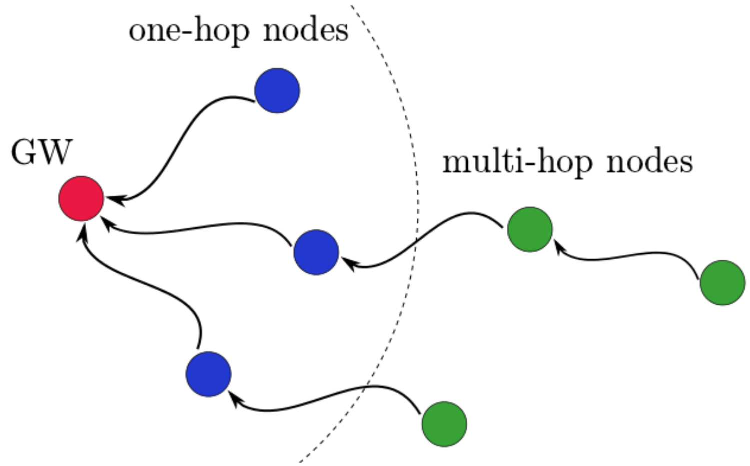


Rate allocation based on the next-hop queue



[1] **Aziz et al., CoNEXT 2009**

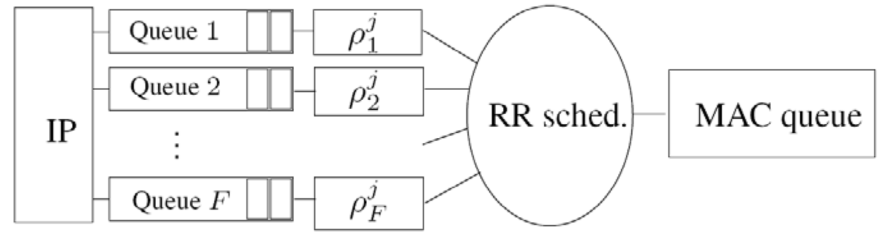
Complete Solution: E&E + EZ

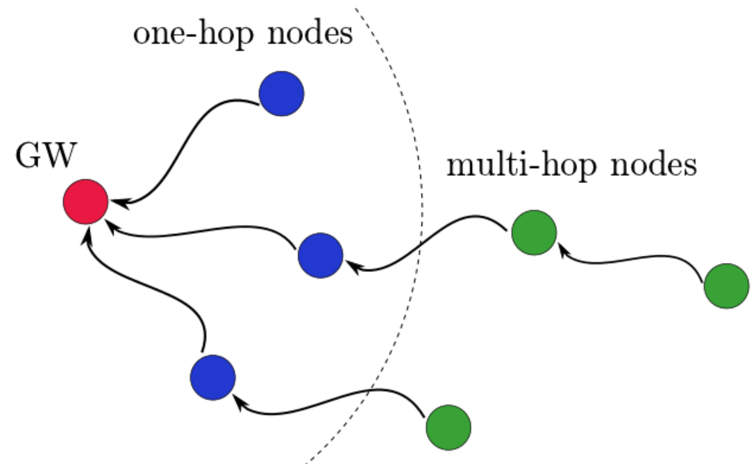


- GW
  - E&E decides the per-flow rate allocation
- One-hop nodes
  - E&E rate-limits one-hop nodes
- Multi-hop nodes
  - EZ-flow rate-limits multi-hop nodes

Practical Implementation

- Based on *Click*[1] with *MultiflowDispatcher*[2]
- Creation of 5 new elements
  - MFQueue
  - MFLeakyBucket
  - EEscheduler
  - EEadapter
  - EZFlow
- Evaluation with
  - Asus routers
  - Ns-3

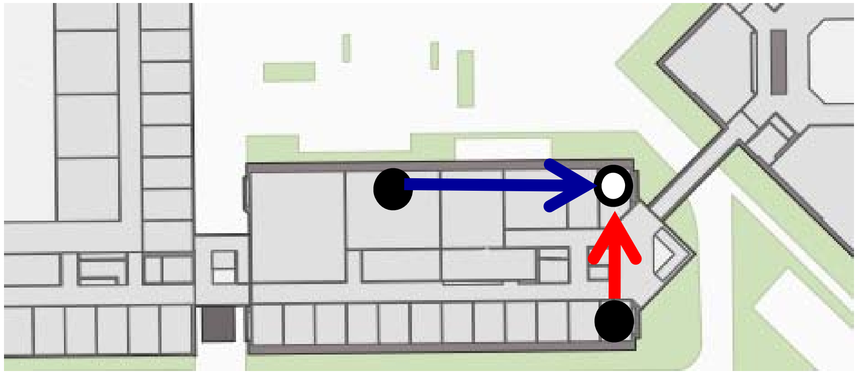




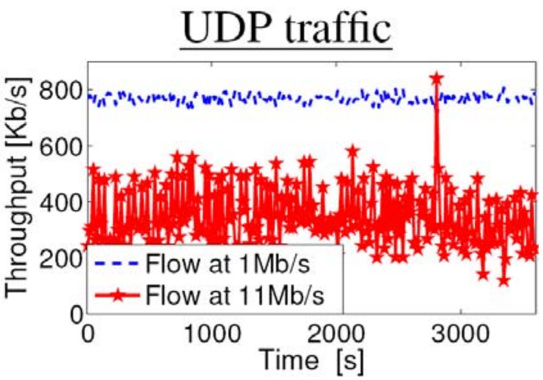
[1] **Kohler et al., Transactions on Computer Systems, 2000**
[2] **Schiöberg et al., SyClick, 2009**

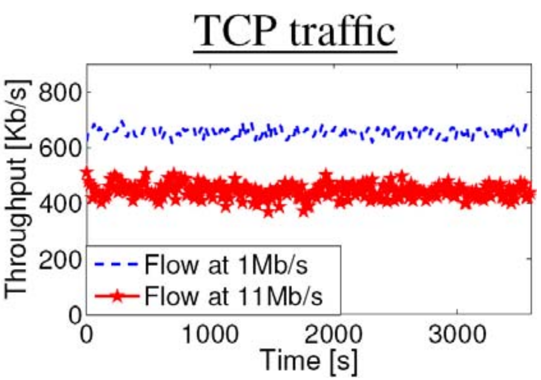
Experimental Results in WLAN

- Deployment map:



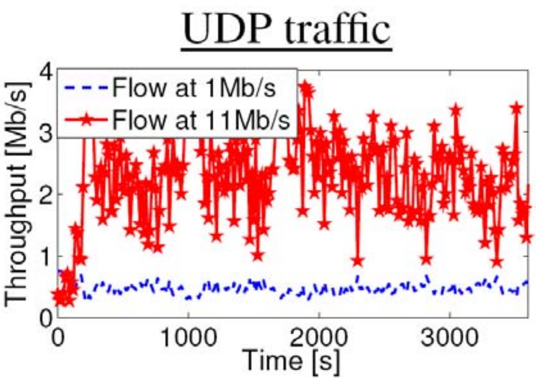
- Without E&E:

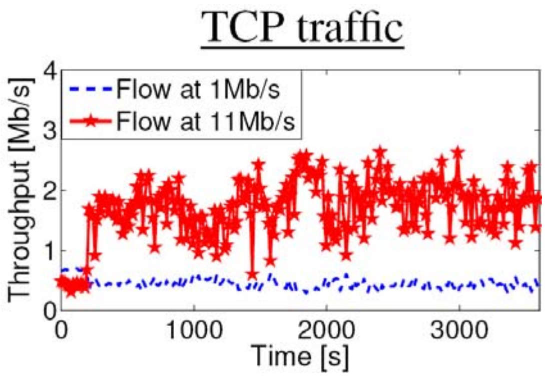




- With E&E:

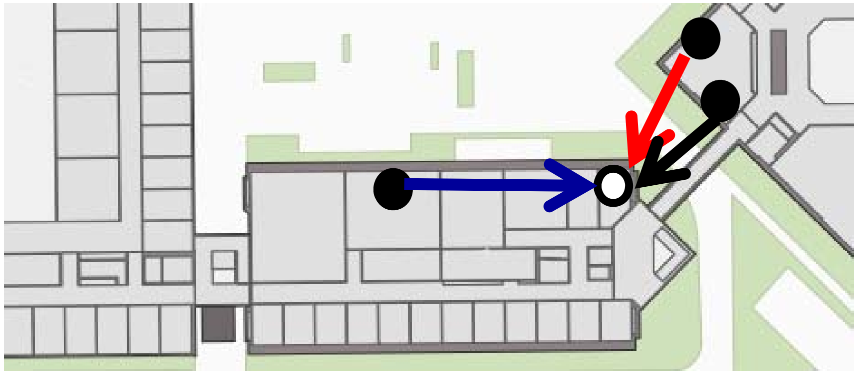
(proportional fairness)



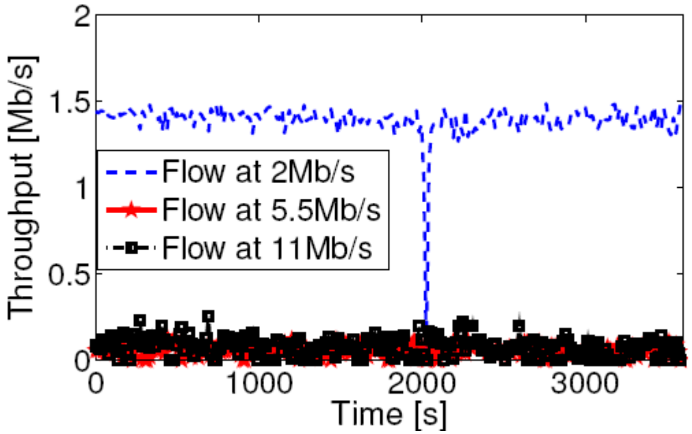


Experimental Results in WLAN

- Deployment map:

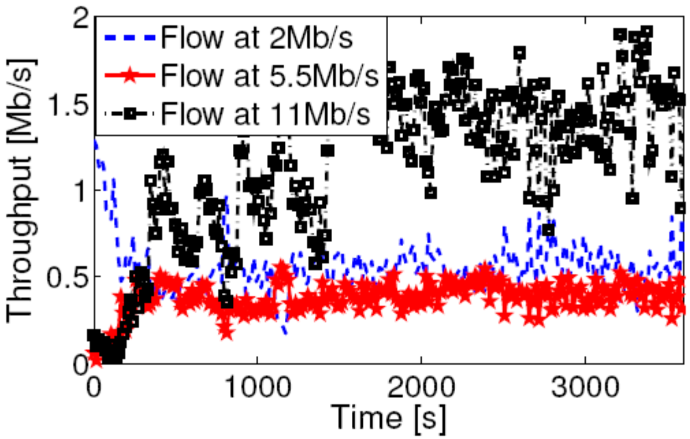


- Without E&E:

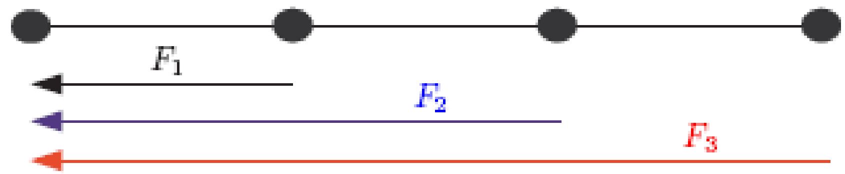


- With E&E:

(proportional fairness)

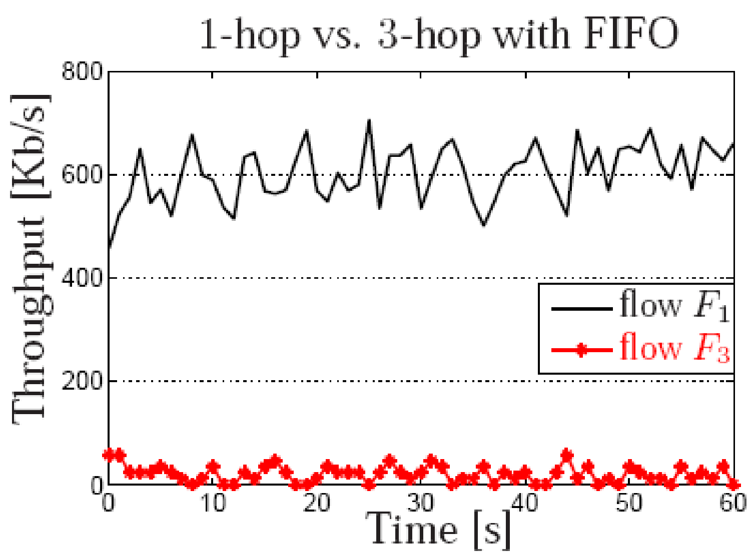


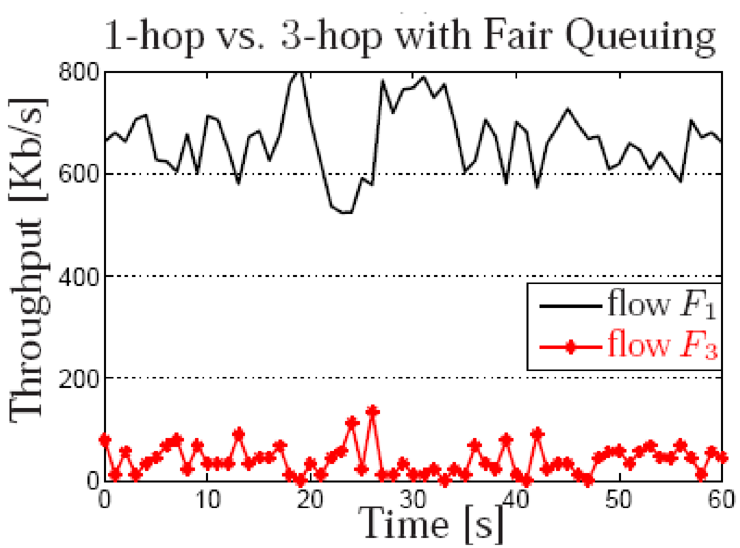
Starvation Problem in Mesh



- Intra-flow congestion control

- Inter-flow fairness
  - Appropriate queuing is needed (Fair Queuing [1])
  - … but it is not enough

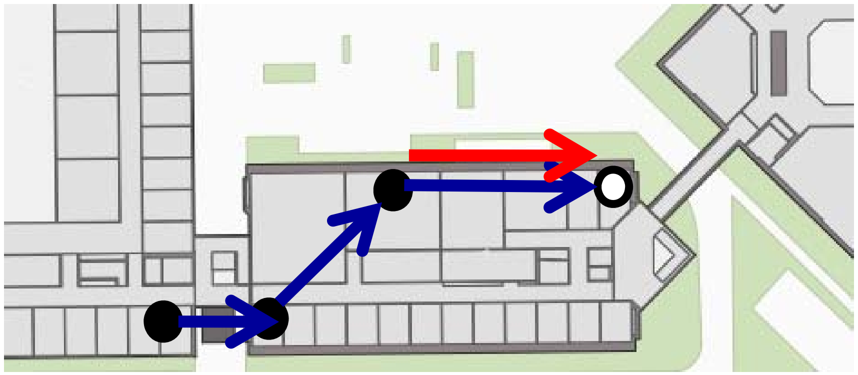




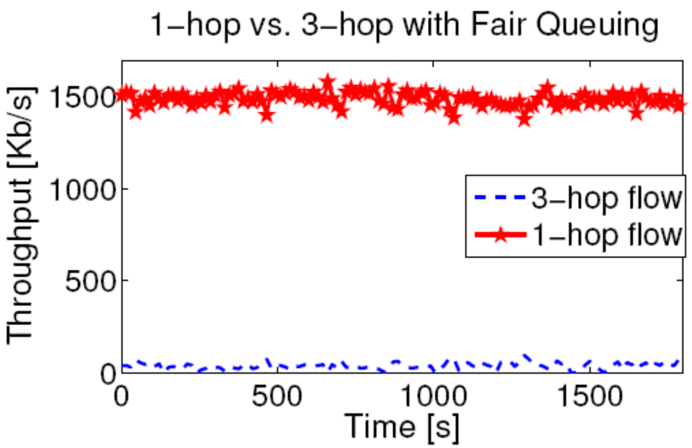
[1] **Demers et al., Sigcomm'89**

Experimental Results in Mesh

- Deployment map:

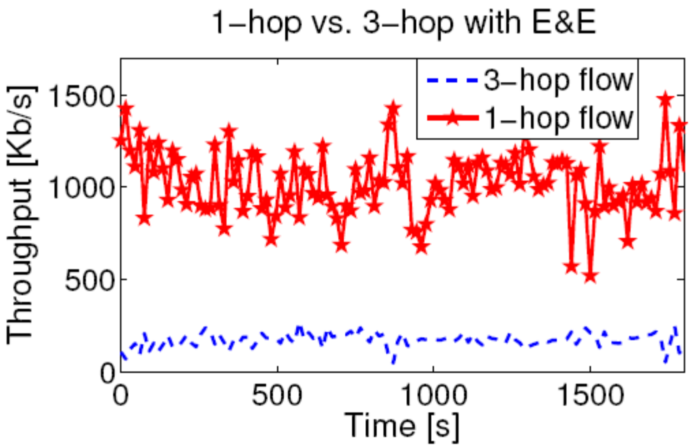


- Without E&E:



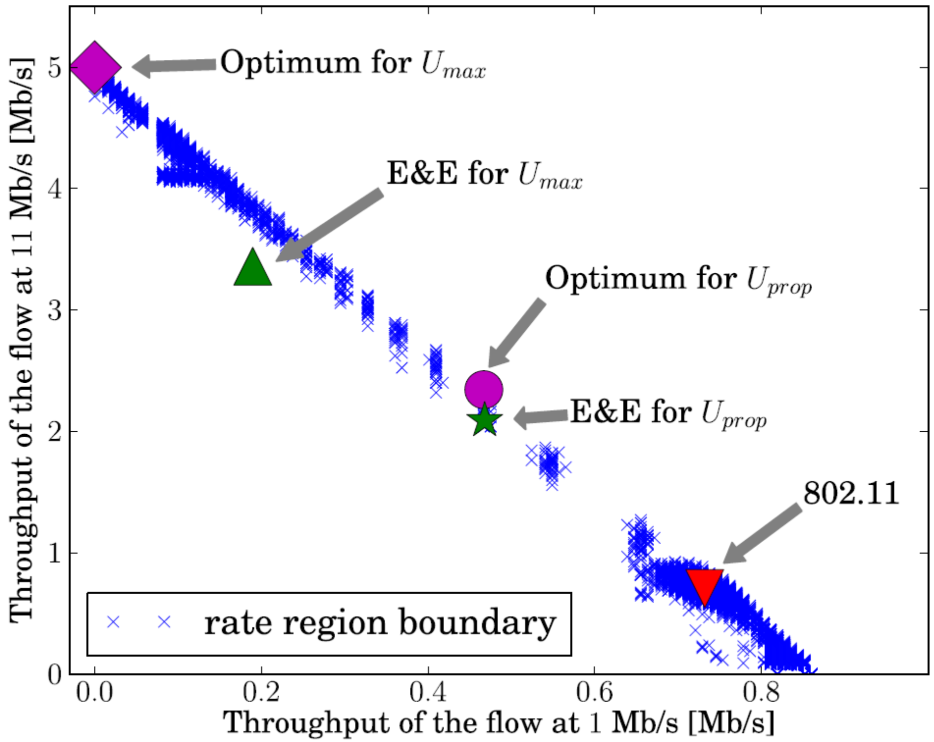
- With E&E:

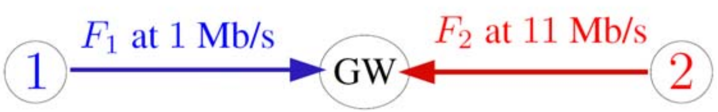
(proportional fairness)



Simulation Results in WLAN

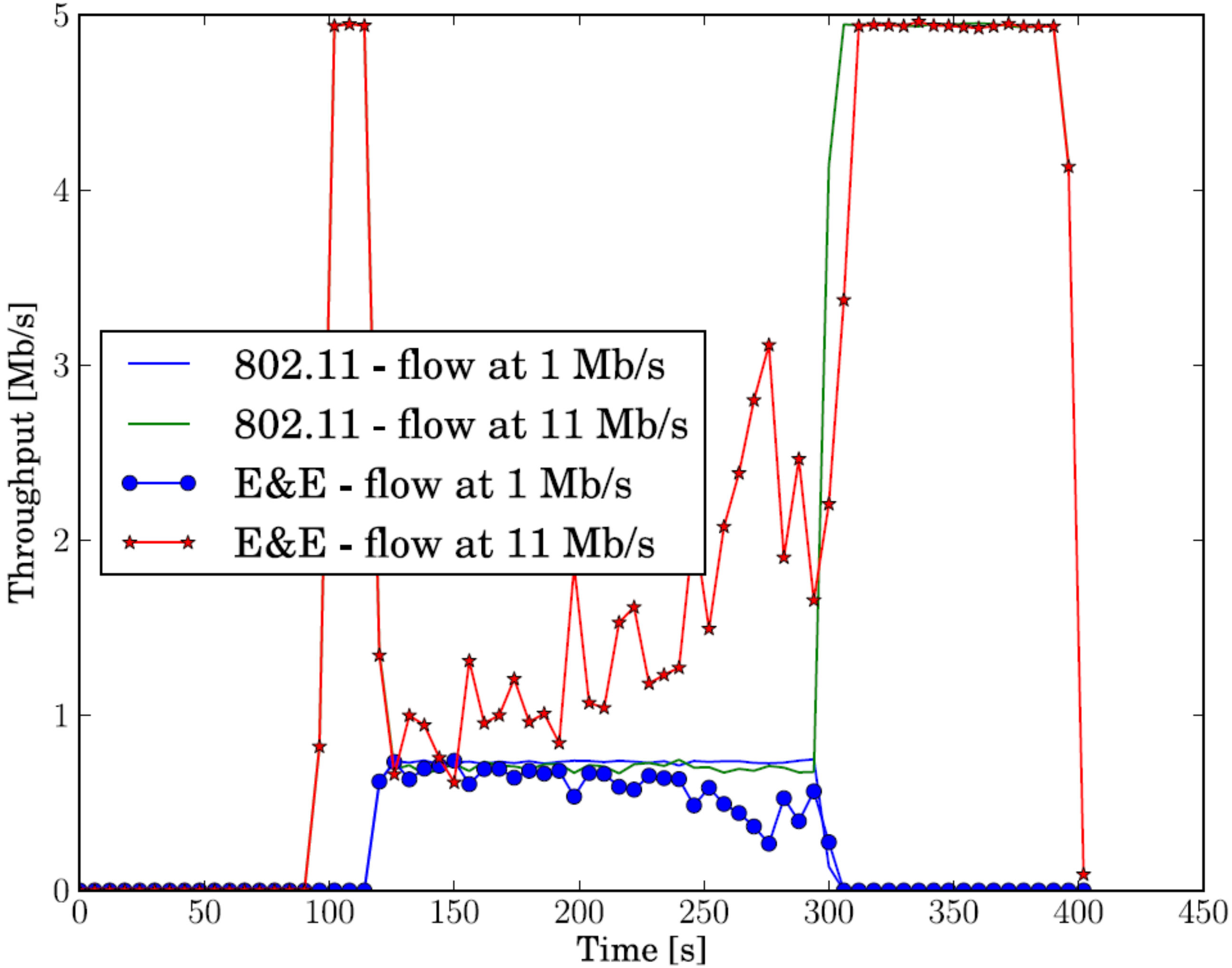
- Ns-3 simulator
  - Re-use of same *Click* elements
  - More controlled environment
  - Possible estimation of capacity
  - Computation of optimum





Simulation Results in WLAN

- Adaptability to time-varying environment
- Cyclic validation of last stable allocation $\vec{r}[n]$



Conclusion

- Wireless networks suffer from
  - Intra-flow problem (e.g., congestion)
  - Inter-flow problem (e.g., unfairness)
  - Time-variability
  - Difficulty/impossibility to characterize the capacity region

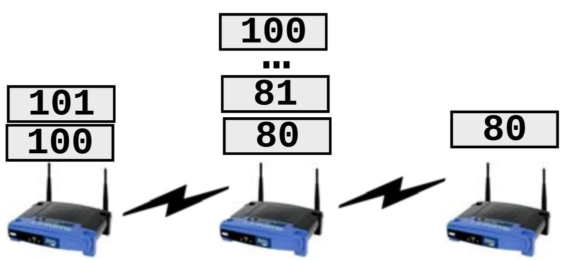
- Need for practical algorithm to maximize a desired utility

- E&E solves the inter-flow problem in WLAN
- Combining E&E and EZ-Flow in mesh
  - Solves both inter-flow and intra-flow problem
  - Avoids network-wide message passing (only one broadcast at GW)

- Future work
  - Study and improve the speed of convergence
  - Analyze new distributions for the *Explore* phase

BOE: Buffer Occupancy Estimation

- Major advantage of EZ-flow
  - The BOE module works passively without any message passing!
  - Uses broadcast nature of the medium
- Advantages of deriving the buffer occupancy implicitly/passively
  - No messaging overhead
  - No header modification/addition
    - Run on current hardware with standard protocol

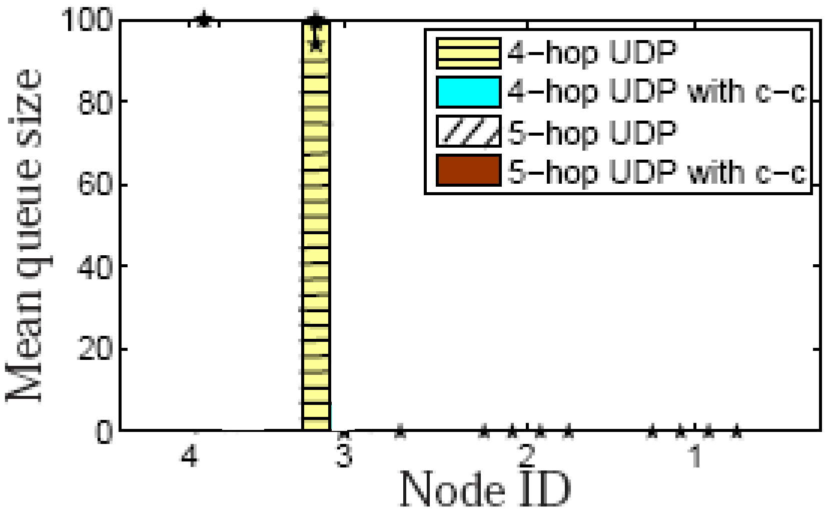


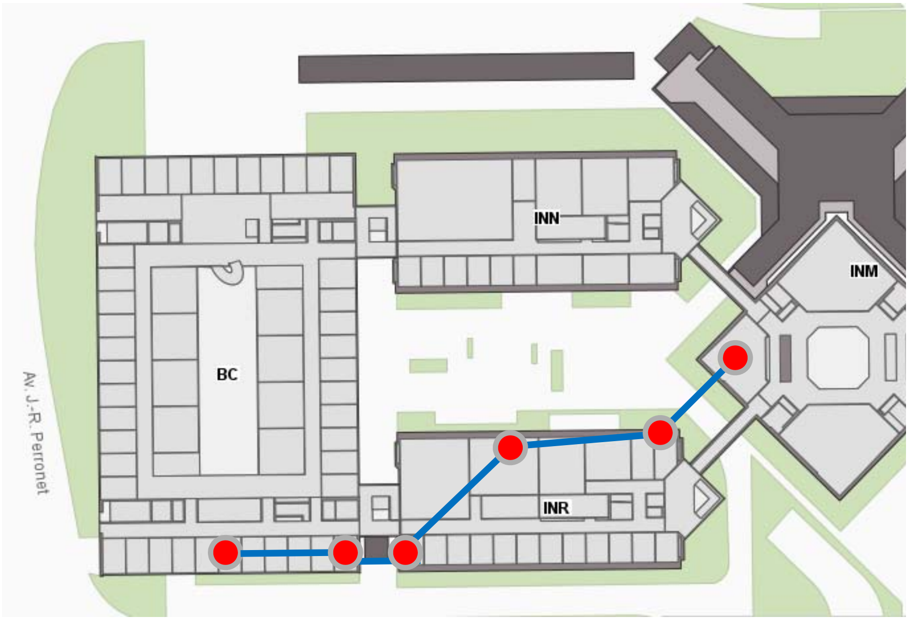
BOE computation: $\boxed{100} - \boxed{80} = 20$

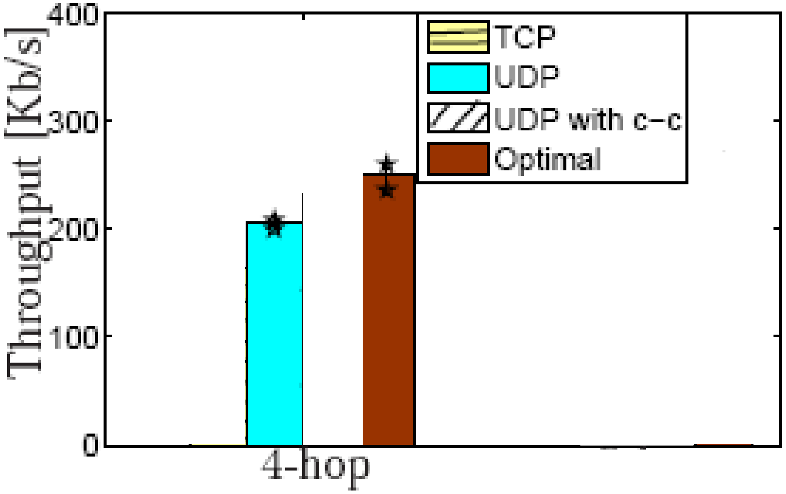
- How to derive the buffer occupancy implicitly?
  - Keep in memory a list of identifiers of the recently sent packets
  - Use the broadcast nature of the medium to monitor the forwarded packet
  - Obtain the current buffer occupancy at successor

Experimental Evaluation of Intra-Flow

- Intra-flow congestion control
  - Stabilizes the queues
  - Increases throughput
  - Not limited by the MAC
  - Higher throughput than TCP







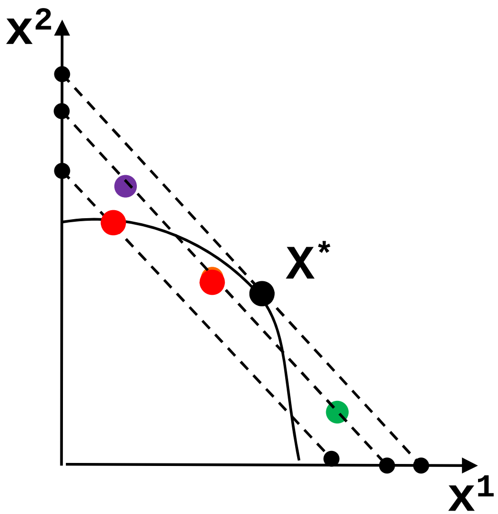
'Enhance & Explore' in WLAN

- Starts from IEEE 802.11 allocation
  - Throughput vector: $x_0 = \rho_0$ [rate allocation]
  - Level set of utility μ_0: $L(x_0, \mu_0)$ $[u(x_0) = \mu_0]$
  - Remember allocation: $r_0 = x_0$
  - Move to next slot: 0 -> 1

- Enhance phase: *(Find the next targeted level set)*
  - If $(x_{i-1} = \rho_{i-1})$

    $\mu_i \leftarrow$ full-size gradient ascent
  - Else

    $\mu_i \leftarrow$ dividing by 2 the size of gradient ascent
  - Go to Explore phase



- Explore phase: *(Find the next allocation)*
  - Pick point ρ_i randomly in $L(\rho_i, \mu_i)$
  - Repeat Explore phase at most **N** times, then move to Enhance phase

- Utility achieved by r(n)
  - Starts from IEEE 802.11
  - Is non-decreasing over time