On the Implementation of Relay Selection Strategies for a Cooperative Diamond Network

Apostolos Apostolaras, Kostas Choumas, Ilias Syrigos, Iordanis Koutsopoulos, Thanasis Korakis, Antonios Argyriou, Leandros Tassiulas

CERTH, The Centre for Research and Technology Hellas.

Dept. of Computer & Communication Engineering , University of Thessaly.

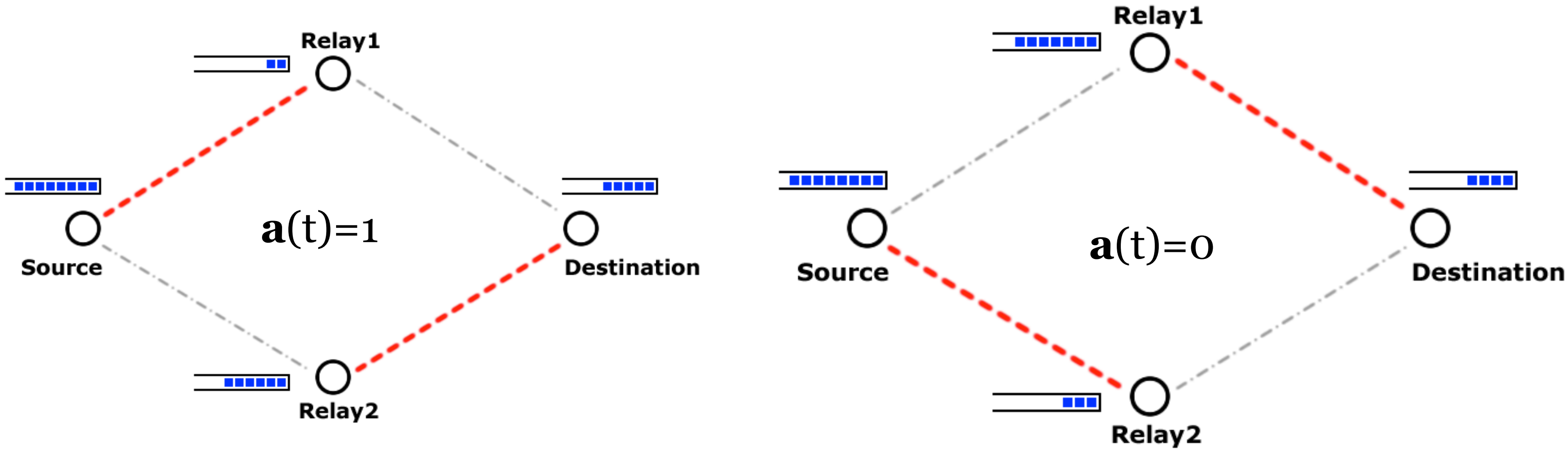
Motivation

- Direct transmissions from Source to Destination may not be the "best" route for information flow.
  - Low QoS due to volatile communication environment
  - Increased Power Consumption
  - Rich interference
  - Delays
  - Degradation in Communication Quality
- **Cooperation:** Exploit different paths through the aid of possible relays for traffic forwarding.
- Aim is to boost networking performance.
  - Increase Throughput
  - Minimize Power Expenditure

Packet-level dynamic relay selection in wireless networks

- Link scheduling for unicast traffic (1 source, 1 destination)
- Packet-level cooperation: opportunistically exploit different paths through relay selection so as to optimize performance
  - Focus on maximizing throughput, minimizing power consumption
  - Traffic dynamics (dynamic packet arrivals in system)
  - Packet queue dynamics (dynamic in- and out-traffic packets from node buffers)
  - Wireless link dynamics (wireless links prone to fading, frequent failures)

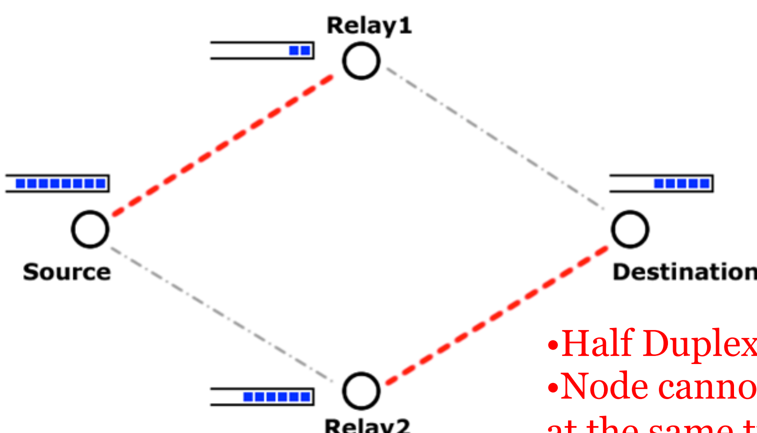
Diamond Network Topology



- Relay Selection: Parrallel Link Activation
- Network controller **a**(t), enables communication on particular links per time slot t attaining a certain optimization goal

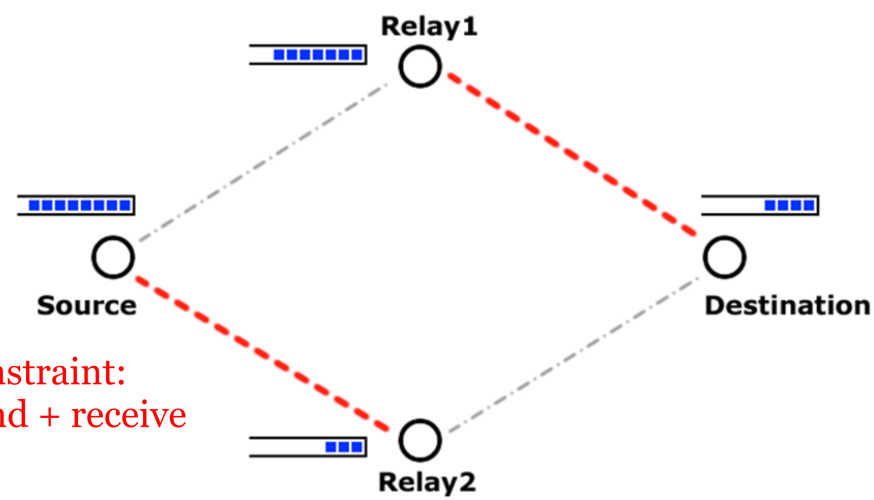
System model

- **System Parameters at time slot t**
  - Node i maintains backlog queue: $Q_i(t)$
  - Exogenous arrivals at node i: $A_i(t)$ (here, only for source node)
  - Transmit power from node i to j: $P_{ij}(t)$
  - Channel State of link k: $S_k(t)$
  - Controller enables schedules: a(t) in {0,1}
  - Service (transmit) rate of link (i,b): $\mu_{ib}(t) = C(P(t), S(t), \mathrm{a}(t))$
  - Evolution of queue of node i: $Q_i(t+1) = \left[Q_i(t) - \sum_b \mu_{ib}(t)\right] + A_i(t) + \sum_a \mu_{ai}(t)$



a(t)=1

•Half Duplex Constraint:
•Node cannot send + receive at the same time



a(t)=0

Optimization Objectives

- Power Minimization
  - Select schedules so as to
    - keep total power consumption low.
    - stabilize queues
- Throughput Maximization
  - Select schedules so as to
    - maximize the total traffic rate end-to-end
    - stabilize the queues

Theoretical Framework

- Framework of Lyapunov optimization
- Define Lyapunov function of queue backlogs: $L(Q(t)) = \sum_i Q_i(t)^2$
- Lyapunov Drift: $\Delta(t) = \mathrm{E}\left[L(Q(t+1)) - L(Q(t)) \mid Q(t)\right]$
  - Expected change of L-function in 1 slot

- **Queue Stability**
- A scheduling policy that maintains queue stability is a policy that minimizes an upper bound on *Lyapunov Drift Δ(t)*

- **Cost function P(.) minimization + Queue Stability**
- A scheduling policy that minimizes cost function P(.) and maintains queue stability is a policy that minimizes an upper bound on *Δ(t) + V P(.)*
- *Lyapunov Drift Δ(t)* + *V* × *Cost Function*

- Here: Cost function related to power minimization

- V is tunable parameter:
  - denotes relative importance of on the problem objective.
  - tradeoff between objective and end-to-end delay.

Problem Formulation

- Minimize upper bound on $\Delta(t) + V\,P(.)$ with respect to scheduling policy a(t) in {0,1}
- Minimize upper bound on $(\Delta(t) + E[P(a(t), S(t)])$

Avg. power consumption

- Scheduling policy that, at each time slot t, takes scheduling decisions based on:
  - Queue state $Q_i(t)$ of nodes in the network
  - Fading state $s_{ij}(t)$ of links (i,j)
  - Transmit power needed for link (i,j) with link quality $s_{ij}(t)$: $P_{ij}(s_{ij}(t))$
  - Power Consumption as a function of link states s(t) and control a(t)

$$P(a(t),S(t)) = a(t)\left[P_{SR_1}(s_{SR_1}(t)) + P_{R_2D}(s_{R_2D}(t))\right] + (1-a(t))\left[P_{SR_2}(s_{SR_2}(t)) + P_{R_1D}(s_{R_1D}(t))\right]$$

Scheduling policy for Throughput Maximization (V=0)

- Transmit rate: $\mu_{ij}(s_{ij}(t)) = \mu_{ij}(t)$, across link (i,j) with link quality $s_{ij}(t)$
- Throughput maximization => only queue stability needed
- At each time slot t, observe differential queue length across links, observe link states
  - If

$$\Delta Q_{SR_1}(t)\mu_{SR_1}(t)+Q_{R_2}(t)\mu_{R_2D}(t) <$$
$$\Delta Q_{SR_2}(t)\mu_{SR_2}(t)+Q_{R_1}(t)\mu_{R_1D}(t)$$

Set network controller a(t)=0 (activate links (S,R2) and (R1,D)

  - Otherwise, a(t)=1 (activate links (S,R1) and (R2,D)
  - Max-weight rule

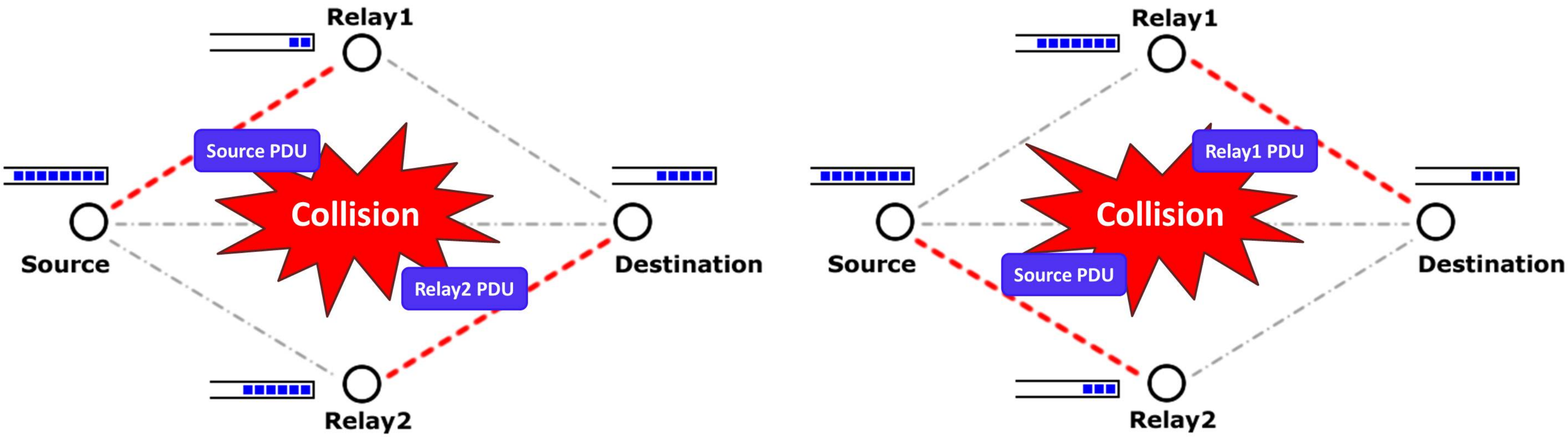
Scheduling Policy for Power Minimization (V≠0)

- V calibrates relative impact:
  - Higher V → controller a(t) tends to select power-efficient schedules.
- This has impact on end-to-end delay: as V grows, queue backlogs also increase in load and buffer congestion increases
- Tradeoff
  - Higher V => less power consumption, but larger end-to-end delay.
  - Smaller V => higher power consumption, and less end-to-end delay.

Implementation Methodology

- **Goal**: Enable parallel transmissions
- **Obstacles**:
  - In commercial wifi products, drivers adopt CSMA to enable collision avoidance when packets collide (*Single Frequency operated networks*)
  - MAC layer prevents us from enabling precise decisions when we want to transmit packets in a time slot
  - This reflects inability to control departure of the packets from the MAC driver queues to be transmitted in the air.
- **Solution:**
  - Relays equipped with two wireless interfaces
  - The two hops in diamond network operate in different channels (to avoid collisions)
  - Enable TDMA to allow for transmissions in particular time slots so as to fortify parallel transmission from colliding
  - Use of Click Modular Router to control the packet transmissions
    - Move Queues to the Network Layer (and control them there), and set the MAC layer queues' maximum length equals to 1

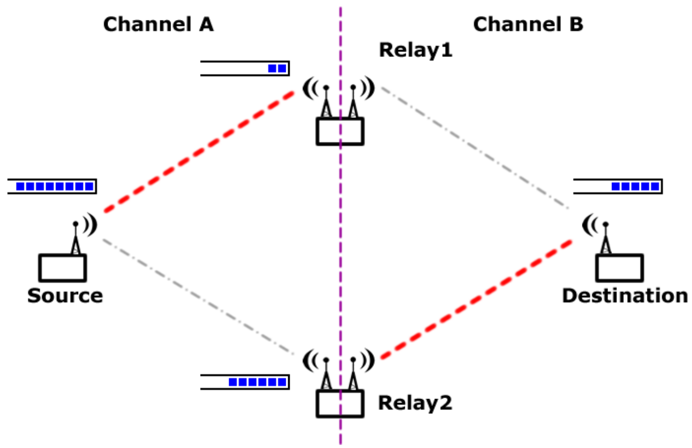
1st Obstacle



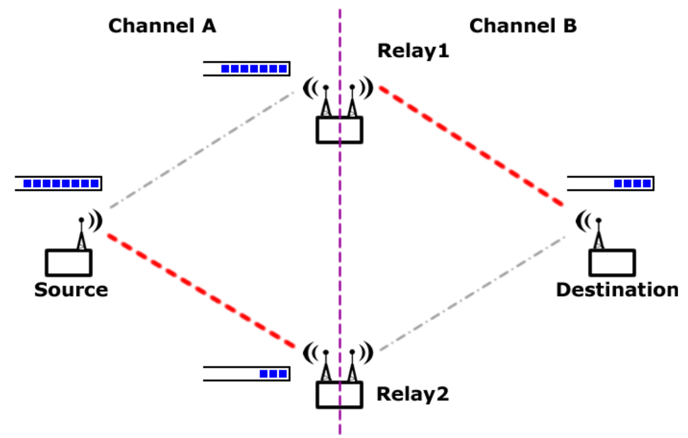
a(t)=1 **a**(t)=0

- In single frequency operated network, packets collide when we enable parallel transmissions
- Even in CSMA, expected throughput benefit from cooperation, will be less due to back-off mechanism
- Solution: Different channels (frequencies for the two parallel links)

Set Orthogonal Transmissions



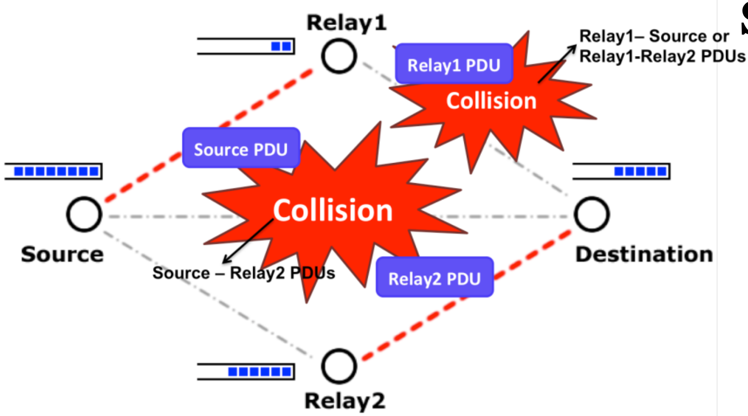
a(t)=1



a(t)=0

- Overcome the problem of collisions in parallel transmission:
  - By letting CSMA operate conventionally,
  - BUT
    - Each transmission per hop enabled on ***different channel*** in order to avoid collisions when parallel transmissions occur.
    - ***Each relay*** is equipped with ***two wireless interfaces*** to enable parallel transmissions on different channels.

2nd Obstacle



Scenario

- Network controller takes a decision to enable the first feasible set { S to R1 and R2 to D} to transmit.
- Some packets were left on the **R1 buffer in the MAC layer** (from previous schedule)

- **We are unable to stop R1 packets from being transmitted**
  - Because we have not control access on the wireless card firmware
  - So orthogonal parallel transmissions are not possible
- Solution: Suppress CSMA

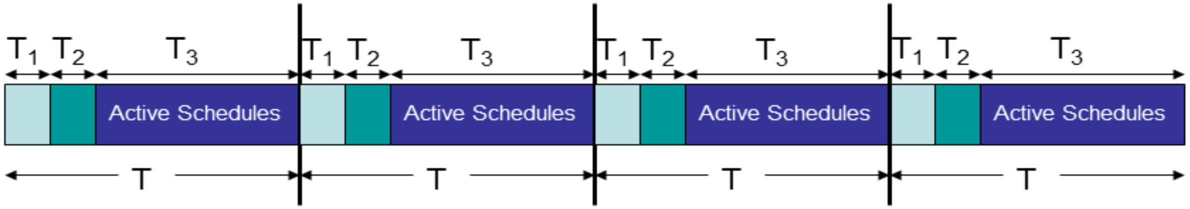
a(t)=1

Suppressing CSMA

- **Make scheduling decisions controllable**
  - *Operate in Sublayer 2.5* where **Click Modular Router** stands
    - Gathering and handling packets is more flexible with Click Modular Router
    - Keep Buffers on Click to store packets
    - Set MAC layer Buffers equals to 1
    - Allow packets entering the MAC layer of a particular node, when this node is scheduled to transmit

- *Create* a **TDMA access** scheme to coordinate transmissions

TDMA frame



Exchange of information about link qualities, queue backlogs …

Control packets that manage the scheduling.

Transmission of data packets.

- T1 interval: Source Node gathers Network State Information from its neighbors. That includes Queue Sizes and Channel Quality Information
- T2 interval: Source takes a scheduling decision according to **max throughput policy** and reports it to the relays by a broadcast message.
  - Set **a**(t)=1, and transmit over the first feasible set.
  - Otherwise, set **a**(t)=0, and transmit over the second feasible set.
- T3 interval: Transmit over the selected schedule set with physical rate μ (t) and power P(t).

Algorithm Implementation

- T1 interval:
  - Relays R1 and R2 report to Source their Queue size and the ETT (Expected Transmission Time) metric for their respective links
  - ETT : an estimation of link state: measured $ETT = \frac{1}{d_f d_r} \frac{B}{S}$

> • d_f and d_r : expected forward and reverse link delivery probabilities
> • S: average packet size
> • B: average packet rate that the rate controller assigns.

- T2 interval:
  - Enable maximum throughput policy and report it to relays by a broadcast message.
  - If $$\Delta Q_{SR_1}(t)\frac{1}{ETT_{SR_1}(t)} + Q_{R_2}(t)\frac{1}{ETT_{R2D}(t)} < \Delta Q_{SR2}(t)\frac{1}{ETT_{SR_2}(t)} + Q_{R_1}(t)\frac{1}{ETT_{R1D}(t)}$$
  - Then a(t)=1, Source broadcasts control message to relays and chooses SR1 and R2D for transmission
  - Else a(t)=0 , Source broadcasts a control message to relays and chooses SR2 and R1D for transmission
- NOTE: (1/ETT) ~ Estimation of the link rate in theoretical algorithm
- T3 interval:
  - Selected schedules activated and packet transmissions are on according to TDMA scheme with physical rate μ(t) and power P(t).

CONECT

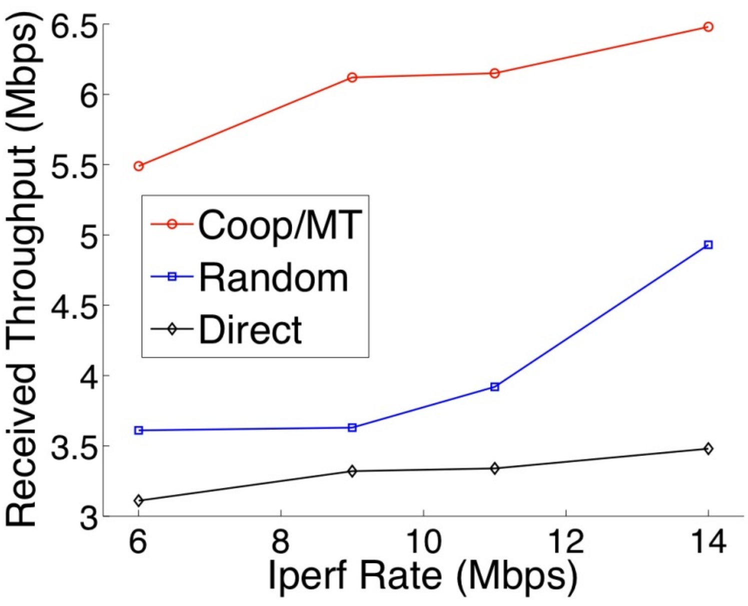
NITLAB NETWORK IMPLEMENTATION TESTBED LABORATORY

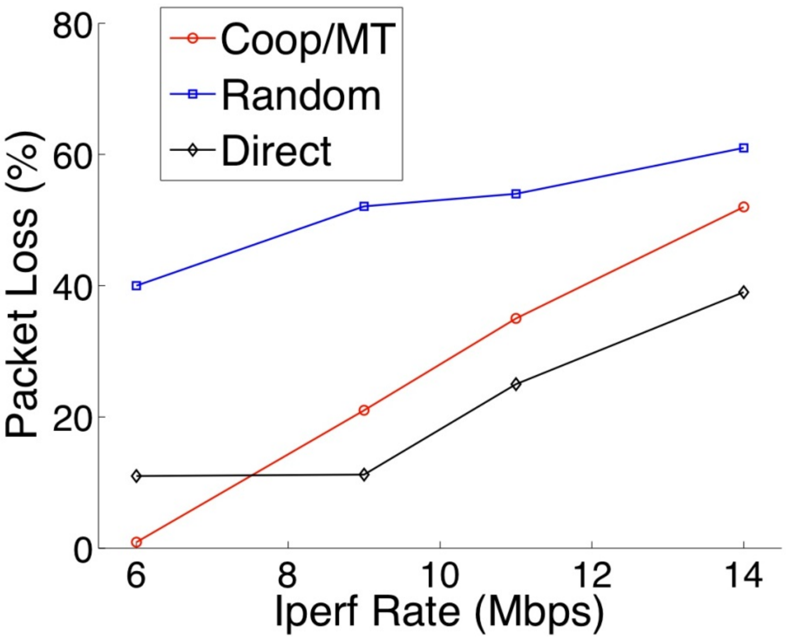
Experiments

- **NITOS Testbed Experimentation**
- **First experiment**
- Implement / compare three algorithms
  - Cooperative relay selection (Lyapunov theory driven)
    - (IEEE 802.11a 1st hop Channel 100, 2nd hop Channel 140)
  - Random relay selection
    - (IEEE 802.11a 1st hop Channel 100, 2nd hop Channel 140)
  - Direct Transmission from Source to Destination (no relay selection)
    - (IEEE 802.11a Channel 120)
    - To emulate a bad fading channel for the direct link from source to destination, we artificially put various AP-STA pairs on the same channel
- **Second Experiment**
  - For different values of max queue length on Click buffer of 2.5 Layer, and for a fixed value of PHY and iperf rate, we measure throughput of the cooperative scheme
  - Better performance (throughput/Packet loss) as max queue size increases

1st Experiment

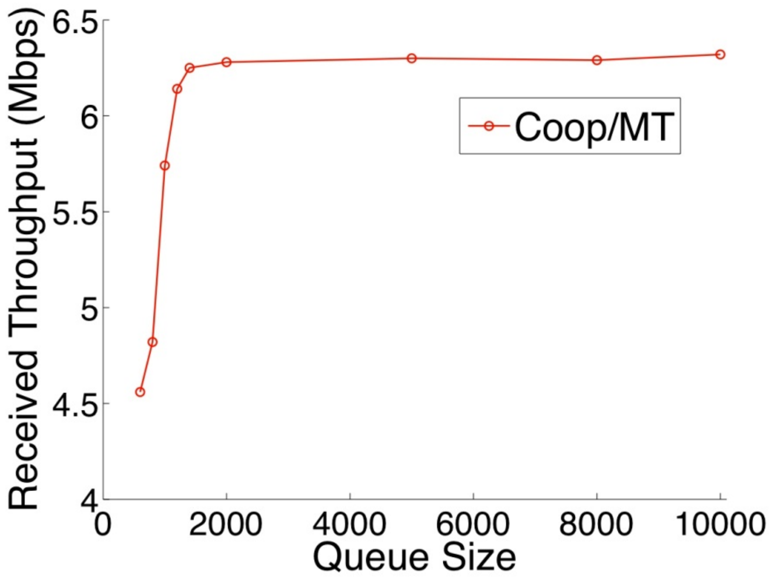
PHY rate was set on 9 Mbps for all nodes

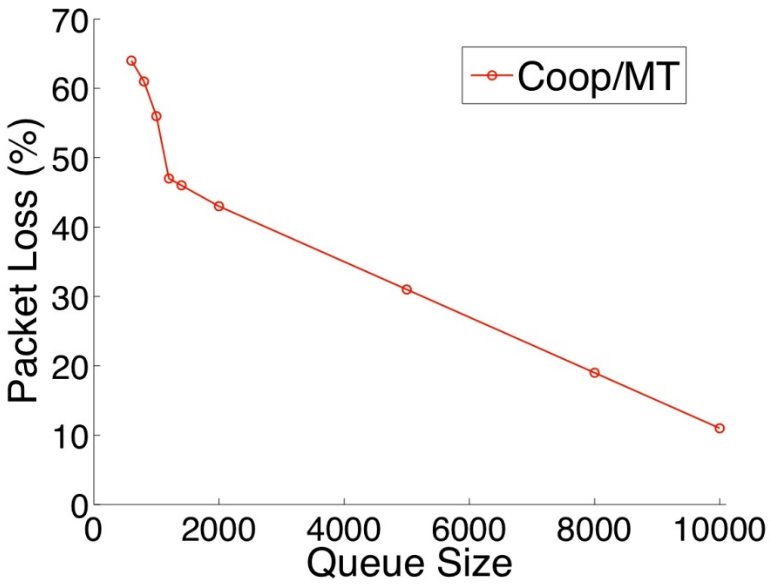




2nd Experiment

PHY rate was set on 9 Mbps for all nodes,
Iperf traffic is set on 14Mbps.





Contribution

- We design and implement a TDMA access scheme for packet forwarding, which is backwards compatible with CSMA enabled commercial devices and it is also effectively applied upon Wi-Fi networks using off-the-shelf equipment.
- We elaborate a centralized network controller in the TDMA frame to enforce scheduling and relay selection policies, relying on Lyapunov optimization.
- We explore performance enhancements of throughput optimal scheduling by implementing centralized networking.
- We evaluated the cooperative maximum throughput solution on the NITOS wireless testbed

Thank you!

- Any Questions?
- http://nitlab.inf.uth.gr
- http://www.conect-ict.eu