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 and Communication Engineering, University of Thessaly
e-mail: {apaposto, kohoumas, ilisirigo, jordan, korakis, anargyr, leandros}@inf.uth.gr

***Abstract*—In this paper, we present an implementation design of a TDMA protocol for the canonical diamond-topology network containing a source, two relays and a destination (single unicast session). Getting inspired by the established Lyapunov-methodology, we propose an online strategy for the relay selection/scheduling problem. In contrast to existing works, we implement this strategy inside the proposed TDMA protocol in order to operate over a CSMA enabled Wi-Fi infrastructure-less network. We elaborate a network controller within the TDMA frame to solve a global optimization problem at each time slot in a centralized manner. In our formulation, we consider the class of scheduling policies that select concurrently a non-interfering subset of links. Our architecture is tailored to achieve the objectives of stabilizing the network and either maximizing throughput or minimizing the total power consumption. Our scheme has been implemented and tested thoroughly through experimentation in the NITOS wireless testbed by exploiting Wi-Fi technology features. The results revealed significant increase in networking efficiency for throughput maximization.**
***Keywords* – Relay Selection, Optimal Centralized Scheduling, TDMA, Wi-Fi Testbed Implementation.**

I. Introduction

In this work, we design a TDMA protocol that elaborates a centralized network controller to activate link scheduling and relay selection decisions towards power minimization or throughput maximization in a cooperative diamond network. We exploit assistance of multiple nodes to cooperate for forwarding a packet when networking conditions are insufficient to favor direct communication. In a previous work [1], we demonstrated the feasibility of this implementation scheme. Here we present a solution that is formed but not limited, to provide an implementation framework for optimization in diamond network topologies through a centralized TDMA access scheme. Relying on the Lyapunov-drift technique from optimization theory [2], we derived a concrete mathematical solution that converges in time and suffices the networking constraints while also optimizes key objectives keeping stability ensured.

An important parameter that we consider is the sensitivity of the proposed solution in key objectives against the total average delay imposed by the congestion on the data queue backlogs. By exploiting the features of the Lyapunov-drift technique, we employ a tunable V parameter that controls the objective goal and can be used to attain a tradeoff between average delay and the objective goal. The objective goal is mapped to a suitable *penalty* function and it is incorporated inside the equation of the drift expression (see [2]). Scheduling actions are taken every slot t to greedily minimize the drift plus the $V \times$ *penalty* expression. This is employed by the proposed TDMA frame design which defines separate periods for networking status collection, message passing for control decisions and actual transmission.

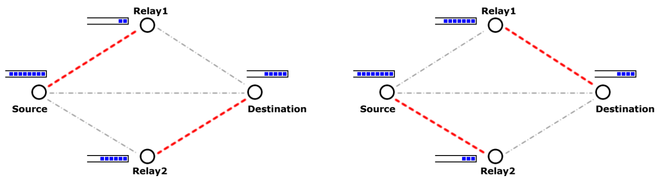
While prior work in wireless cooperative networks has shown that enhancement in communication can be achieved in terms of throughput performance, low delay, low power consumption and QoS guarantees, this work realizes the potential of centralized networking for taking online scheduling decisions to enforce optimization objectives such as throughput maximization or power minimization. Intellectual merit of this work relies on the following contributions:

- 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 seek to obtain desired tradeoffs between networking performance efficiency metrics, such as power consumption (or system throughput) vs. networking delay.
- We evaluated the cooperative maximum throughput solution on the NITOS wireless testbed [3].

II. Related Work

In this section, we describe previous works in the literature, studying the relay selection problem in cooperative wireless networks. The seminal information-theoretic works of Cover *et al.* [4] and Schein *et al.* [5] introduced a study for the capacity of relay networks. The conventional three node relay model was also studied by Van der Meulen in [6]. In [7] Berry *et al.*, study cooperative communication models that incorporate stochastic traffic arrivals for multiple sessions as well as the related queueing dynamics in all network nodes.

Guan *et al*. in [8] investigate the problem of joint spectrum management and relay selection for a set of sessions in order to utilize an interference limited infrastructure-less network. Madan *et al*. in [9] study a cooperative wireless network where



(a) 1st feasible action set, $\mathbf{a}(t) = 1$. (b) 2nd feasible action set, $\mathbf{a}(t) = 0$.

Fig. 1: Relay Selection policy: Controller $\mathbf{a}(t)$ enables communication on particular links each time. Red doted lines indicate active schedules.

a set of nodes cooperate to relay in parallel the information from a source to a destination using a *decode-and-forward* (DF) approach. A network coding approach for the same problem set was studied in the work of *Lucani et al.* in [10]. The influential work of Tassiulas in [11] established a framework for indicating throughput optimal scheduling in radio networks by introducing the well known backpressure algorithm that exploits queueing backlogs as system state information, to derive an optimal scheduling policy. As an enhancement of that work, Neely in [12] developed a dynamic control strategy for minimizing energy expenditure in a time varying wireless network with adaptive transmission rates. The aforementioned works set up the foundation for creating a concrete framework [2], for resource allocation in several wireless network types. This framework provides a solid presentation of the theory of Lyapunov optimization applied on queueing networks and have formed the foundations for the work presented in this paper. We exploit this framework for enabling an extensible elaboration for cooperative relaying networks.

Apart from theoretical works in the literature, there are several studies considering cooperative implementations in infrastructure-less wireless networks. To the best of our knowledge the first implementation were the works of Liu *et al.* in [13] and Korakis *et al.* in [14], [15] where authors designed a cooperative MAC protocol, simple and backward compatible with the legacy IEEE 802.11 standard. In those works, each low data rate node selects either direct transmission or assisted-relay transmission by utilizing a *CoopTable* where it stores potential helper/relays rate information. Neighboring nodes exchange their rate tables so as to automatically decide whether a packet will take longer to transmit between two nodes directly or via a cooperative relay assistance. In another work [16], Nikolyenko *et al.* presented a hardware-independent data link layer design for cooperative retransmission support to the Linux kernel wireless SoftMAC implementation (mac80211). Laufer *et al.* in their work [17] design a throughput-optimal backpressure architecture for wireless multi-hop networks. A mesh network is transformed into a wireless switch, where packet routing and scheduling decisions are made by a backpressure scheduler elaborating a TDMA access scheme in a centralized manner. Although the proposed framework was evaluated in a Wi-Fi testbed and revealed significant performance improvements, an important drawback of this work is the inability to be seamlessly applicable on commercial Wi-Fi devices, since it was implemented hard-coded on the firmware of a particular vendors Wi-Fi card.

III. Problem Formulation

This section focuses on the development of an optimization formulation for diamond wireless networks that enables a switching between relays for link activation over feasible scheduling sets, in order to achieve an optimization objective either *Power Minimization* or *Throughput Maximization*.

A. Network Model

We consider the two-hop diamond network as depicted in Fig. 1 consisting of a source S, two relays R_1, R_2 and a destination D. Time t is slotted. We assume the existence of a system controller which lies in the source S and it is denoted by $\mathbf{a}(t)$. Its role is to enable the two feasible scheduling actions, as shown in Fig. 1, in a *centralized* manner. Thus, the controller can take the following values $\mathbf{a}(t) \in \{\mathbf{1}_{\{SR_1,R_2D\}}, \mathbf{0}_{\{SR_2,R_1D\}}\} = \{1,0\}$ relying on the collected system state information from neighboring nodes. Networking dynamics change over time and system state denoted by $\mathcal{S}(t) = \{\mathcal{S}_{SR_1}(t), \mathcal{S}_{SR_2}(t), \mathcal{S}_{R_1D}(t), \mathcal{S}_{R_2D}(t)\}$ along with queuing dynamics cause the system controller to select one scheduling action over the other, by evaluating a particular metric-rule which reflects an optimization objective. Each node i maintains a buffer in the network layer for storing incoming packets that are waiting for transmission over the wireless links. Let $Q(t)$ represent the current number of packets (or number of bits) in the queue which evolves according to the following equation:

$$Q_i(t+1) = [Q_i(t) - \sum_b \mu_{ib}(t)]^+ + A_i(t) + \sum_a \mu_{ai}(t), \quad (1)$$

where $[\cdot]^+ = \max(\cdot, 0)$ and $A_i(t)$ are the exogenous arrivals on node i at time slot t. The transmission rate $\mu_{ab}(t) = \mu_{ab}(\mathcal{S}_{ab}(t))$ in the link (ab) at slot t depends on the link channel state condition $\mathcal{S}_{ab}(t)$. We assume that the channel state $\mathcal{S}(t)$ is known at the beginning of each time slot t and remains constant over its duration, but it can be variable throughout time slots. We also make the following assumptions:

- Packet injection $A(t)$ in the network takes place only on the source node S.
- Without loss of generality, power consumption $P_{ab}(t)$ in node a for transmitting to node b is spent only during packet transmissions while receptions are *free of charge*.

B. Optimization Framework

Given the topology in Fig. 1 and considering the channel state and queue size variations, the objective is to designate a relay selection/scheduling policy that determines unicast transmission activation with the separate goals of either **1)** *minimizing the total power consumption* or **2)** *maximizing the total throughput* of the network.

Solution: The scheduling policy rule is educed by minimizing the bound on the Lyapunov drift (see [2], [11], [12]) expression given in Eq. 2 with respect to $\mathrm{a}(t)$. The solution follows simply, and the network controller indicates power efficient schedules by setting $\mathrm{a}(t) = 1$ and selecting $\{SR_1, R_2D\}$

for activation, when the expression inside brackets is negative ($[\cdot] < 0$), otherwise it sets $\mathrm{a}(t) = 0$ and activates $\{SR_2, R_1D\}$.

$$\begin{aligned}\mathrm{a}(t)\Big[& V\left(P_{SR_1}(t) + P_{R_2D}(t) - P_{SR_2}(t) - P_{R_1D}(t)\right) \\ & - Q_S(t)\left(\mu_{SR_1}(t) - \mu_{SR_2}(t)\right) + Q_{R_1}(t)\left(\mu_{R_1D}(t) + \mu_{SR_1}(t)\right) \\ & - Q_{R_2}(t)\left(\mu_{R_2D}(t) + \mu_{SR_2}(t)\right)\Big] < 0 \end{aligned} \tag{2}$$

The drift formula incorporates a tunable V parameter for calibrating the networking delay (as induced by the packet congestion in the buffer queues) against the optimization objective, which is reflected through a suitable *penalty* function (power consumption). This enables a desirable *performance-delay tradeoff* that we seek to attain by suitably tuning the V parameter. The higher the V value, the more the rule goes away from **backpressure** policy [11], sacrificing throughput for power reduction. The less the V value, the more throughput is achieved, however, with a significant expense on power consumption. Next, we briefly analyze two cases and we give in Table I the corresponding activation metrics-rules.

Power Minimization – ($0 < \mathbf{V} < \infty$): The objective is to keep the total power consumption low while also stabilizing the queues. Hence, we seek to activate schedules and relay selection accordingly. We impose a per-node power constraint that reflects the maximum transmission power for a single node. The analysis suggests that the proposed policy can achieve a total power expenditure arbitrarily close to the optimal value (for sufficiently large V), at the cost of increased queue congestion and a corresponding delay.

Throughput Maximization – ($\mathbf{V} = \mathbf{0}$): An appropriate selection of the tuning parameter V, makes the initial problem of Power Minimization to be reduced to Throughput Maximization. By observing the power minimization formulation, if we set $V = 0$, we get a max weight scheduling decision algorithm. The objective now is to select schedules so as to maximize the total traffic rate of the cooperative network subject to the same constraints (power, schedules, etc.) as in the power minimization problem.

IV. Wi-Fi Implementation

In this section, we consider a Wi-Fi implementation on the diamond topology of Fig. 1, by designing a centralized scheduling and relay selection algorithm. The adopted CSMA access scheme in Wi-Fi networks prevents us from the direct appliance of this algorithm. We firstly seek to design an implementation methodology, that describes the techniques that will be used to solve particular problems arising when parallel transmissions are activated, and latterly to present the complete Wi-Fi solution. Overcoming the Wi-Fi limitations includes a design of a TDMA frame and a method to suppress CSMA. The novelty of this implementation lies in its generality to enforce activation for scheduling and parallel transmission in Wi-Fi operated networks.

A. Implementation Methodology

a) Packet Collision during Parallel Transmission in Single Frequency Operated Networks: In a single frequency operated network, packets collide when parallel transmissions are enabled. This is depicted in Fig. 2 where we observe packets colliding when parallel transmissions (from S to R_1 and from R_2 to D) or (alternatively, from S to R_2 and from R_1 to D) are activated. Even enabling CSMA for collision avoidance, the expected throughput benefit from cooperative relaying will be lower due to the back-off mechanism. In order to overcome that obstacle the solution that we adopt, is to separate the diamond network in two hops, where the links belong to the first hop (links from source to relays) and the links belong to the second hop (links from relays to destination) operate in different channels. In this way, parallel transmission in the frequency domain is orthogonal, see Fig. 3.

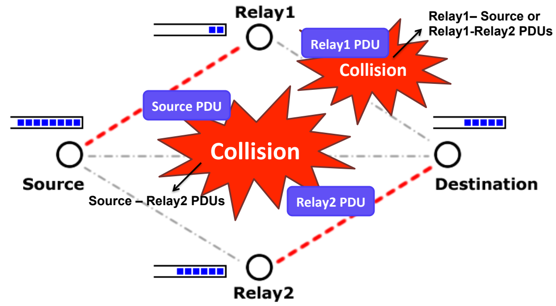


Fig. 2: Packet collisions occurring when parallel transmission are activated and due to limited control access on the wireless card firmware.

b) Packet Collision due to Limited Access on the Wireless Card Firmware: Let us consider the following scenario where the network controller takes a decision to activate two links for transmission (SR_1 and R_2D), after an active transmission session on the other set of links (SR_2 and R_1D). The problem that occurs is the following: Some packets might be left on the R_1 buffer in the MAC layer queue (from the previous schedule) and since there is no control of the transmission in the wireless card firmware, a collision with a packet coming from R_1 with any packet being transmitted in the active links (either SR_1 or R_2D) can occur (see Fig. 2 - up-right collision).

c) Solution: In order to be able to prohibit the aforementioned collisions, we firstly must be able to precisely define the time periods where each scheduling decision is active and secondly to be able to stop packets that belong to different schedules rather than the nominal activated to be transmitted. In order to achieve the first goal, we design a TDMA access method where we orchestrate the transmission periods of each schedule with accuracy. For the second goal, we exploit the features of the Click Modular Router framework [18], and we choose to operate in OSI sublayer 2.5 where this framework stands. We maintain buffers on Click in order to store packets, due to the flexibility that Click offers for packet gathering and handling. Moreover, we set the MAC layer queue size equals to one and we allow packets entering the MAC layer of a particular node only when this node is selected for transmission.

B. Design of a TDMA Frame

We elaborate a TDMA scheme and we activate it upon the Wi-Fi operation that uses carrier sense multiple access (CSMA) for accessing the shared medium. Per hop, the same frequency/channel is used among users and the medium access

TABLE I: SCHEDULING ACTIVATION RULES.

$(\mathbf{V}=\mathbf{0})$	***Throughput-Maximization***	$(\mathbf{0}<\mathbf{V}<\infty)$	***Power-Minimization***
if	$(Q_S(t)-Q_{R2}(t))\,\mu_{SR_2}(t)+Q_{R_1}(t)\mu_{R_1D}(t) >$ $(Q_S(t)-Q_{R1}(t))\,\mu_{SR_1}(t)+Q_{R_2}(t)\mu_{R_2D}(t)$	***if***	$V\left(P_{SR_1}(t)+P_{R_2D}(t)-P_{SR_2}(t)-P_{R_1D}(t)\right) - \left(Q_S(t)-Q_{R_1}(t)\right)\mu_{SR_1}(t)-Q_{R_2}(t)\mu_{R_2D}(t) + \left(Q_S(t)-Q_{R_2}(t)\right)\mu_{SR_2}(t)+Q_{R_1}(t)\mu_{R_1D}(t)>0$
then	$\mathbf{a}(t)=0$, **Activate** $(SR_2),(R_1D)$ Schedules.	***then***	$\mathbf{a}(t)=0$, **Activate** $(SR_2),(R_1D)$ Schedules.
else	$\mathbf{a}(t)=1$, **Activate** $(SR_1),(R_2D)$ Schedules.	***else***	$\mathbf{a}(t)=1$, **Activate** $(SR_1),(R_2D)$ Schedules.

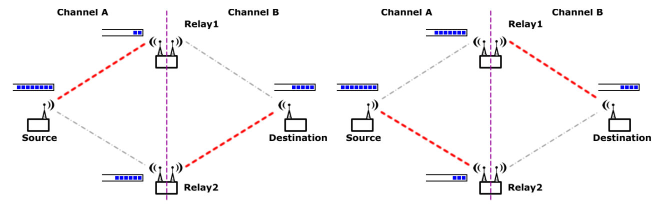


Fig. 3: Feasible schedules are denoted with red lines. Each transmission per hop is enabled on different channel in order to avoid collisions when parallel transmissions occur.

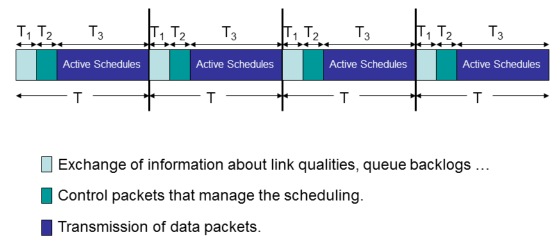


Fig. 4: TDMA frame for the cooperative diamond network.

is divided into different time slots. Thereby, active users transmit using their pre-allocated time slot.

For enabling the proposed TDMA access scheme, we need a *centralized* control mechanism to gather network statistics such as temporal backlog loads or power consumption levels, so that the source S to obtain this information coming from the rest nodes inside the network. This control mechanism along with a message-passing mechanism about schedules, are employed inside a TDMA frame structure that provides accurate timing information to all nodes in the network.

A TDMA frame structure showing a data stream divided into frames, and those frames divided into time intervals, is depicted in Fig. 4. In T_1 interval, the mechanism for gathering network statistics is employed and the source S is getting updated with the network state. In sequence, the next interval T_2 marks the broadcast transmission, from the source S to the relays R_1 and R_2, and it is about the control packets that activate the appropriate schedules. The decision about the schedules has been pre-computed as soon as the S was full aware about the updated network state and before T_2 starts. After successful reception of the control broadcast signal, relays are getting synchronized and decide to transmit or not, in T_3.

Since T_2 and T_3 depend on the success in T_1, a potential error or delay in T_1, that may lead to a certain information unavailable to the centralized cotroller, will not severely inflict system's performance, since the network will continue to operate relying on the previous scheduling decision. However, this will cause on the long term, a convergence-delay to the optimization objective, as the number of errors increases, but with a signigicant robustness profit in terms of ensured queueing stability, a characteristic which comes from the Lyapunov drift optimization that allows the controller for capturing changes on the networking conditions each time slot t and adapt on the networking dynamics accordingly.

Time duration of the intervals in the frame structure are pre-defined in the system configuration setup and can be calibrated considering the volatility of the networking conditions. The denser, the operations occurring in T_1 and T_2 are triggered, the more accurate the scheduling activations are and the network converges faster to the problem objective. However, this implies increased overhead and latency. Thereby, the TDMA access scheme comes with a cost due to synchronization issues along the time periods. Despite the aforementioned cost, the merit of this scheme is the actual performance improvement by realizing the centralized proposed policies, that is also verified by the experimentation results. Moreover, this TDMA scheme is not limited to a sole diamond network and can be extended to apply on expanded WLAN topologies. The diamond pattern is repeated among combinations of nodes, belonging on a WLAN that form canonical diamonds (each diamond can be activated in a round-robin way for avoiding extra collisions), as shown in Fig. 5, hence, achieving also a transmission range extension.

C. CSMA Protocol Suppression

Recall the diamond network of Fig. 1 consisting of a source S, two relays R_1, R_2 and a destination D. In order to be able to assume that links are interference limited and impose the constraint that, at any time slot t, only one of the two sets of links (shown in (left-side) or (right-side)) can be activated, we need to operate the wireless network in different channels per hop and equip relays with two wireless interfaces, in order to avoid packet collissions. Thus, first hop links ($S \to R_1$ or $S \to R_2$) use channel A and second hop links ($R_1 \to D$ and $R_2 \to D$) use channel B. In Fig. 3, this setup is illustrated. Moreover, considering the case where collisions might be occurred when packets from an active schedule (i.e. links SR_1 and R_2D) collide with packets from a previous active schedule (i.e. from link R_1D) due to the inability of controlling the wireless card firmware, we face this obstacle by moving the queues and the storing of packets to the Sublayer 2.5, using Click Modular router. Moreover, we set the maximum length size (capacity in terms of number of packets) of the MAC layer queues of each node equals to one, so that to be able to suppress the inability of controlling packet transmissions from the MAC queue.

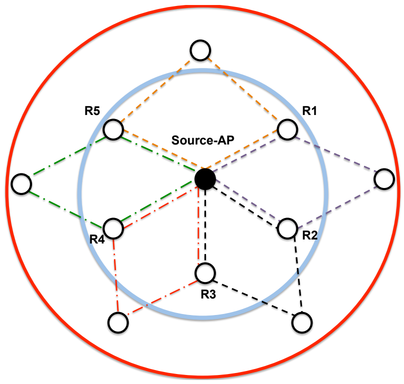


Fig. 5: The solid red circle indicates the extended transmission coverage range in a WLAN. Wireless nodes belonging on a WLAN act as forwarders, forming canonical diamonds to assist networking when conditions are insufficient to benefit direct transmission on the cell-edge nodes. Each diamond can be activated i.e., in a round-robin way, one by one, for avoiding extra collisions.

D. *Algorithm Implementation*

The implementation of *maximum-throughput* scheduling policy by designing a system architecture that enables parallel transmission in the diamond network is given. Our system architecture exploits packet handling features of the Click Modular Router [18] and the ath9k driver [19]. A significant point is that we use the ETT value [20] in order to acquire an estimation for the transmission rate and to capture link quality. The system design is *transparent* to each policy and the difference relies on the implementation of each rule to achieve the different objectives. Let us recall the TDMA access scheme described in the previous paragraph and analyze the execution steps in each period:

T_1 Period, Relays report to the source S the length of their queue size and the ETT value of each link where they belong. ETT metrics are enabled and gathered by using Click and their value is given by the following formula: $\text{ETT} = \frac{1}{d_f d_r}\frac{B}{L}$, where d_f and d_r are the expected forward and reverse link delivery probabilities (product of these two is the probability of a successful acknowledged transmission), B is the bandwidth and L the packet data size.

T_2 Period, The source S has already gathered the required information (queue sizes and the ETT metrics) in the previous period T_1. The source node calculates the maximum throughput policy by evaluating $\Delta Q_{SR_1}(t)\frac{1}{\text{ETT}_{SR_1}(t)} + Q_{R_2}(t)\frac{1}{\text{ETT}_{R_2D}(t)} < \Delta Q_{SR_2}(t)\frac{1}{\text{ETT}_{SR_2}(t)} + Q_{R_1}(t)\frac{1}{\text{ETT}_{R_1D}(t)}$. If the aforementioned condition is true then controller sets $\mathrm{a}(t) = 0$ and this scheduling decision is sent to relays from the source S by broadcasting a control message reporting that the active schedules for the next time slot will be SR_2 and R_1D. Otherwise, controller sets $\mathrm{a}(t) = 1$ and source broadcasts a control message to relays denoting the active schedules for the next time slot to be SR_1 and R_2D.

T_3 Period, Selected schedules are activated and packet transmissions are enabled according to the TDMA scheme. Per packet rate and power configuration is enabled through tweaking the Radiotap header [21]. Each packet, upon the scheduling transmission decision is configured with the appropriate power and rate level. Those values are the system parameters used for by the activation rule to produce the optimal scheduling/relay selection decision.

V. Experimental Results

For the performance evaluation of the proposed policy aiming at maximizing throughput efficiency, we conducted 2 different experimentation setups. We used NITOS [3] testbed facility, located in the University of Thessaly in order to select a diamond network topology and perform the experiments.

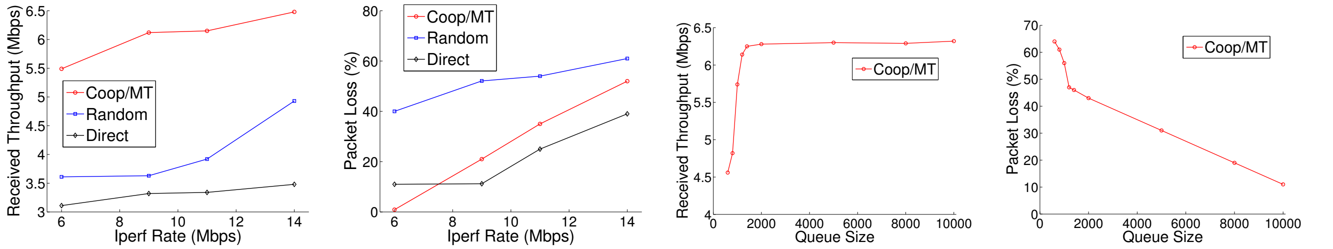
A. 1^{st} *Experiment*

In this experiment setup, we compare the received throughput and the packet losses of the cooperative *maximum-throughput* (Coop/MT) policy where $V = 0$, against a *random* and a *direct* (source transmits directly to the destination) scheduling policies. We configured the links on the first hop (second hop) to operate on channel 100 (140) respectively, in IEEE 802.11a mode. For a static configuration of the PHY rate that was set on 9 Mbps, we collected measurements regarding the received throughput, when we injected UDP application traffic on different rates (6, 9, 11 and 14 Mbps) by using *Iperf* tool. Moreover, for the same experimentation setup, we collected the packet losses. The TDMA access scheme period was set on 100 ms, and each experiment session last for 1 min. We repeated this experiment 10 times to take average values.

Experimental Inference: Results show that Coop/MT policy achieves higher throughput results for different values of the *Iperf* rate. Specifically, in the direct transmission scenario, we selected an experimentation setup of rich interference where nearby wireless networks were configured to operate on the same channel as the source/destination pair in the diamond network. So we assumed a setup, where direct transmission from S to D does not benefit throughput efficiency. Results collected are illustrated in Fig. 6.a. For the packet losses, we observed that when we injected higher application rate in the *Iperf*, we received higher packet losses. This is expected, since the PHY rate configured on 9 Mbps acts as a bottleneck, when we use higher application *Iperf* rates, causing higher packet drops and losses. Packet losses on Coop/MT are lower comparing to packet losses on random scheduling policy, and this justifies the performance efficiency of the Coop/MT scheduling policy. Fig. 6.b shows a graphical representation for packet losses. The lower percentages of packet losses when we use direct transmission is explained by the fact that in the case of Coop/MT and random selection policies, we employed the TDMA framework built in Click [18] for packet controllability and forwarding that injects extra synchronization overhead and latency. However, in the *direct* case this framework is missing.

B. 2^{nd} *Experiment*

In this experiment setup, we aim at identifying the impact of the buffer queue size on the throughput performance and the packet losses considering the *maximum-throughput*



(a) Received Throughput.

(b) Packet Losses.

(c) Received Throughput.

(d) Packet Losses.

Fig. 6: **1st *Experiment***: **(a)** Received Throughput at Node D and **(b)** Packet Losses for Different Values of *Iperf* for Coop/MT, Random and Direct Transmission Policies, **(b)** Packet Losses for Different Values of *Iperf* for Coop/MT, Random and Direct Transmission Policies, **2nd *Experiment***: **(c)** Received Throughput and **(d)** Packet Loss in Coop/MT Policy, for Different Values of Network-Layer (Click Buffer) Queue Size.

(Coop/MT) policy. We kept the MAC layer queue size equals to 1 (one), as a means to enforce the suppression of CSMA protocol and to avoid undesirable collisions. We set a constant PHY rate at 9 Mbps, and the *Iperf* rate at 14 Mbps and we collected the received throughput at the destination D and the packet losses for different sizes of the network-layer queue, spanning from 600 packets to 10000 packets capacity size. Each experiment session duration was also set on 1 min and repeated 10 times for taking average values.

Experimental Inference: We observe a significant rise in throughput as the queue length size increases (up to a critical point) while also at the same time the packet loss percentage drops. This behavior is expected due to the fact that the more the queuing capacity storage availability, the more the capability of receiving higher amounts of traffic is satisfied, having impact in higher throughput and lower packet drops. After the critical point, despite the increase in the buffer queue size the received throughput is saturated, since the buffer queue size does not affect the throughput performance any more, however packet loss percentage drops significantly down, since larger buffer capacity storage aids in improved maintenance and robust packet transmissions.

VI. Conclusion

In this paper, we presented an implementation design for a TDMA protocol about optimal relay selection in cooperative diamond networks. The significant merit of this work is the exploitation of the potential of centralized scheduling through TDMA to achieve optimal performance efficiency and the elaboration of a network controller within a TDMA frame. Our implementation was based on modifying the ath9k [19] wireless driver and exploiting the features of Click Router [18] while the architecture is backwards compatible on commercial Wi-Fi devices and not limited in the presence of the CSMA protocol. Another significant point is that our design can be effectively applied on more complicated topologies where various diamond networks can span in WLAN topologies. The focus of our future work includes the testing of the power-optimal relay selection and scheduling algorithm and its application in more complicated wireless topologies.

Acknowledgment

The research leading to these results has received funding from the European Community's 7th Framework Programme, CONECT (FP7/2007-2013) under grant agreement n^{o}257616.

References

[1] A. Apostolaras, K. Choumas, I. Syrigos, G. Kazdaridis, T. Korakis, I. Koutsopoulos, A. Argyriou, and L. Tassiulas, "A Demonstration of a Relaying Selection Scheme for Maximizing a Diamond Network's Throughput," in *TRIDENTCOM*, 2012, pp. 408–410.

[2] L. Georgiadis, M. J. Neely, and L. Tassiulas, *Resource Allocation and Cross-layer Control in Wireless Networks*. Foundations and Trends in Networking, 2006.

[3] NITOS, "Network Implementation Testbed using OpenSource Software." [Online]. Available: http://nitlab.inf.uth.gr/.

[4] T. Cover and A. Gamal, "Capacity Theorems for the Relay Channel," *Information Theory, IEEE Transactions on*, vol. 25, Sep 1979.

[5] B. Schein and R. Gallager, "The Gaussian Parallel Relay Network," in *Information Theory, Proc. IEEE International Symposium on*, 2000.

[6] E. C. van der Meulen, "Three-Terminal Communication Channels," *Adv. Appl. Probab.*, vol. 3, pp. 120–154, 1971.

[7] E. Yeh and R. Berry, "Throughput Optimal Control of Cooperative Relay Networks," *Information Theory, IEEE Transactions*, Oct. 2007.

[8] Z. Guan, T. Melodia, D. Yuan, and D. A. Pados, "Distributed Spectrum Management and Relay Selection in Interference-limited Cooperative Wireless Networks," ser. MobiCom '11. ACM, 2011, pp. 229–240.

[9] R. Madan, N. Mehta, A. Molisch, and J. Zhang, "Energy-Efficient Cooperative Relaying over Fading Channels with Simple Relay Selection," *Wireless Communications, IEEE Transactions on*, vol. 7, Aug. 2008.

[10] D. Lucani, F. Fitzek, M. Medard, and M. Stojanovic, "Network Coding for Data Dissemination: It Is Not What You Know, But What Your Neighbors Don't Know," in *WiOPT*, 2009.

[11] L. Tassiulas and A. Ephremides, "Stability Properties of Constrained Queueing Systems and Scheduling Policies for Maximum Throughput in Multihop Radio Networks," *Automatic Control, IEEE Trans.*, 1992.

[12] M. Neely, "Energy Optimal Control for Time Varying Wireless Networks," in *INFOCOM*, vol. 1, March 2005.

[13] P. Liu, Z. Tao, S. Narayanan, T. Korakis, and S. S. Panwar, "CoopMAC: A Cooperative MAC for Wireless LANs," *Selected Areas in Communications, IEEE Journal*, vol. 25, no. 2, pp. 340 –354, February 2007.

[14] T. Korakis, Z. Tao, S. Makda, B. Gitelman, and S. Panwar, "It Is Better to Give Than to Receive: Implications of Cooperation in a Real Environment," in *Proc. of IFIP-TC6 conf.*, ser. NETWORKING'07, 2007.

[15] T. Korakis, Z. Tao, S. Singh, P. Liu, and S. Panwar, "Implementation of a Cooperative MAC Protocol:Performance and Challenges in a Real Environment," *EURASIP Wireless Comm. and Networking Journal*, 2009.

[16] V. Nikolyenko and L. Libman, "Coop80211: Implementation and Evaluation of a SoftMAC-based Linux Kernel Module for Cooperative Retransmission," in *WCNC, 2011 IEEE*, March 2011, pp. 239 –244.

[17] R. Laufer, T. Salonidis, H. Lundgren, and P. Le Guyadec, "XPRESS: a Cross-layer Backpressure Architecture for Wireless Multi-hop Networks," ser. MobiCom '11. ACM, 2011, pp. 49–60.

[18] E. Kohler, R. Morris, B. Chen, J. Jannotti, and M. F. Kaashoek, "The Click Modular Router," *ACM Trans. Comput. Syst.*, vol. 18, Aug. 2000.

[19] ATH9k, "The Ath9k Wireless Driver." [Online]. Available: http://linuxwireless.org/en/users/Drivers/ath9k/.

[20] R. Draves, J. Padhye, and B. Zill, "Routing in Multi-radio, Multi-hop Wireless Mesh Networks," ser. MobiCom '04. ACM, 2004.

[21] Radiotap, "The Radiotap Header for 802.11 Frame Injection and Reception." [Online]. Available: http://www.radiotap.org