A. Moraleda-Soler, B. Coll-Perales and J. Gozálvez. "Link-Aware Opportunistic D2D Communications: Open Source Test-bed and Experimental Insights into their Energy, Capacity and QoS Benefits", Proceedings of the 11th IEEE International Symposium on Wireless Communication Systems 2014 (ISWCS 2014), 26-29 August 2014, Barcelona (Spain). DOI: 10.1109/ISWCS.2014.6933425

# Link-Aware Opportunistic D2D Communications: Open Source Test-bed and Experimental Insights into their Energy, Capacity and QoS Benefits

Alejandro Moraleda-Soler, Baldomero Coll-Perales and Javier Gozalvez
UWICORE Laboratory, Miguel Hernandez University of Elche
Avda. de la Universidad, s/n, 03202, Elche (Spain)
amoraleda@umh.es, bcoll@umh.es, j.gozalvez@umh.es

*Abstract*— **Device-to-Device (D2D) communications can efficiently support the growth in mobile data traffic by offloading part of the traffic from the cellular infrastructure. D2D communications are influenced by the propagation conditions between mobile devices that depend on the antenna heights, presence of obstacles, and mobility of devices. Analytical and simulation studies have shown that link-aware opportunistic transmission schemes can improve the reliability and efficiency of D2D communications. This paper goes a step further and experimentally demonstrates the energy, capacity and Quality of Service (QoS) benefits of link-aware opportunistic D2D communications. This is done through a novel test-bed that is here presented and released open source to the community. The developed test-bed modifies the Linux kernel so that it is independent of the particular hardware being used and can be utilized with Commercial Off-The-Shelf (COTS) equipment.**

***Keywords— Device to Device; D2D; link-aware; cross-layer; opportunistic; transmission; Linux kernel; experimental.***

## I. INTRODUCTION

Device-to-Device (D2D) communications have been proposed to offload cellular data traffic and improve the system's capacity and energy consumption [1]. To achieve the expected benefits, D2D communications should be able to reduce the impact of unstable and poor link quality conditions that can be produced under challenging propagation conditions between mobile devices [2]. Link adaptation and power control can help reduce such impact, but can result in the use of low-bandwidth and high power transmission modes under fast and continuous changes in the D2D link quality conditions [3]. Other studies propose the use of cross-layer opportunistic schemes that pause communications under unreliable and low-efficiency link quality conditions[1,2]. One significant contribution on this topic is presented in [4] where the authors propose a buffer-aided relaying protocol with adaptive link selection for delay constrained and unconstrained transmissions. The proposed protocol schedules transmissions based on the channel state, and significantly improves the throughput compared to conventional relaying. Another relevant contribution is reported in [5] where the authors propose a scheduler that opportunistically schedules data transmissions based on the channel conditions. The proposed scheduler is based on optimal stopping theory, and uses channel state information and the knowledge of the transmission deadline. The study demonstrates that significant energy benefits can be obtained with opportunistic communications without degrading the end-user QoS.

The studies reported in [4] and [5] highlight the benefits of opportunistic transmission schemes through analytical and numerical evaluations. It is then necessary to experimentally confirm such benefits for which this paper presents what is to the authors' knowledge the first experimental implementation and evaluation of link-aware opportunistic D2D communications. The study is based on IEEE 802.11 technologies due to the current lack of LTE-based D2D hardware. It is important noting though that the 3GPP considers the use of both Long Term Evolution (LTE-Direct) and IEEE 802.11 (WiFi Direct) technologies for D2D communications [6]. In addition, opportunistic transmission schemes will benefit D2D communications independently of the communications standard employed.

The contributions of this paper are threefold: 1) the paper presents a novel test-bed that implements link-aware opportunistic D2D transmissions by modifying the Linux kernel, which allows the use of COTS (Commercial Off-The-Shelf) equipment and an implementation that is independent of the communications hardware utilized; 2) the developed software test-bed is released open source to the community; 3) the presented test-bed is used to conduct the first experimental analysis of the energy, capacity and QoS benefits that can be achieved with link-aware opportunistic D2D communications.

## II. EXPERIMENTAL WIRELESS RESEARCH

There is an increasing trend in the wireless community towards experimental research using real platforms and prototypes. The use of such platforms fosters the transfer of research, and avoids simplifications usually done in analytical and simulation studies. Experimental research can thus provide the means for a deeper understanding of protocols, and a more realistic evaluation of their performance. Experimental wireless research has been fostered by the emergence of Software Defined Radio (SDR) platforms such as USRP or WARP. The programmability of SDR platforms provides considerable benefits for prototyping advanced

---

[1] Pausing D2D communications can introduce certain transmission delays. However, according to the latest Cisco's global mobile data traffic forecast, many of the services and applications that are driving the growth of data traffic can be deemed delay tolerant (e.g. updates to social networking, emails, firmware and software updates or cloud services).

[2] These opportunistic transmission schemes pause D2D transmissions based on the quality and efficiency of D2D links. This is significantly different from DTNs that address the lack of continuous network connectivity using the store, carry and forward paradigm.

wireless communications and networking protocols. SDR platforms are thus commonly employed for experimental cross-layer wireless research. For example, [7] utilizes an SDR platform to investigate and test a joint cross-layer and cooperative PHY/MAC layer that utilizes intermediate relays to improve quality. Despite its benefits, it is challenging to achieve with SDR platforms similar performance levels to that obtained with commercial devices due to the software processing delays [7]. An alternative is the use of open source drivers (e.g. madwifi, ath9k, HostAP or iwlwifi). Using this approach, [8] implements and evaluates two new QoS support schemes for IEEE 802.11e that serves stations based on the priority of the group they belong to or the priority of the contents of the packets to be transmitted. Compared to SDR platforms, the use of open source drivers maintains the performance levels of commercial hardware, but can limit the capacity to modify the normal operation of protocols depending on how open is the access to the HAL (Hardware Abstraction Layer). In fact, no implementation of opportunistic transmission schemes using open source drivers has been found in the literature. The use of open source drivers also tights a particular implementation to the communications hardware controlled by the drivers. Such 'coupling' can be avoided when the implementation is done modifying open source operating systems such as Linux. In this case, the implementation is not tight to a particular hardware and commercial performance levels can still be maintained. This approach is used in [9] where the authors utilize the Click software architecture to implement and test a novel scheduling algorithm for 802.11 that prevents nodes experiencing poor channel conditions from monopolizing the wireless medium. In [10], the authors modify the Linux kernel in an Android device to enable the IBSS (or ad-hoc) mode. The authors also create sockets between the kernel space and the user space to monitor the link quality conditions and check the stability and the performance of the IBSS network. Considering its advantages, this study opted for the option of modifying the default 802.11 implementation under the Linux kernel to implement what is, to the authors' knowledge, the first experimental test-bed for link-aware opportunistic D2D communications.

## III. LINK-AWARE OPPORTUNISTIC D2D TEST-BED

The objective of this study is to experimentally demonstrate the energy, capacity and QoS benefits of link-aware opportunistic D2D communications. To this aim, a link-aware opportunistic transmission scheme is implemented in an experimental test-bed. The scheme determines when the data transmission between two devices should be paused/re-started in order to improve performance and reliability, and reduce energy consumption and channel occupancy/utilization. The implemented scheme has been designed to experimentally demonstrate the benefits of link-aware opportunistic D2D communications, but does not pretend to be optimum. The scheme is implemented modifying the default 802.11 implementation in the Linux kernel, in particular its Medium Access Control (MAC) sub-layer (<linux/net/mac80211>). The implemented scheme is released open source at http://www.uwicore.umh.es/opportunities/Opp-D2D.html.

### A. *Link-aware opportunistic transmission scheme*

The implemented test-bed considers two IEEE 802.11 mobile devices that exchange data, management and control packets. These packets are necessary to maintain and manage the network connectivity, and measure the D2D link performance. It is important noting that the opportunistic transmission scheme is only applied to the transmission of data packets, and not to the transmission of management and control packets. The two mobile devices can hence continuously monitor the link quality even if the transmission of data packets is paused due to the inefficiency of D2D links.

The implemented opportunistic transmission scheme uses the Received Signal Strength Indicator (RSSI) link quality metric to schedule transmissions. This metric was chosen since it can be experimentally obtained at low computational cost in most wireless devices. In particular, the implemented scheme estimates the link quality conditions between two mobile devices using the measured RSSI levels of all received packets (data, management, and control). The scheme is aimed at improving the efficiency and performance of D2D communications by avoiding unnecessary (and inefficient) transmissions when the link quality conditions are not good. Such conditions are here identified by an RSSI threshold ($RSSI_{thr}$). The scheme pauses D2D transmissions between devices when the measured link quality conditions are below $RSSI_{thr}$. To account for fast variations of the link quality, D2D communications should not be paused when a single RSSI measurement is below $RSSI_{thr}$, but instead when the average RSSI ($RSSI_{avg}$) measured for the last *Nb_Rx* received packets is below $RSSI_{thr}$. It is also important to consider the fact that D2D Line-Of-Sight (LOS) conditions are likely to be intermittently lost because of the presence of vehicles, obstacles, etc., which will result in temporary drops of the RSSI that do not correspond to a disconnected D2D link. The implemented scheme allows then a number (*Nb_BelowThr*) of consecutive $RSSI_{avg}$ values below $RSSI_{thr}$ before pausing the data transmission between two mobile devices. If a D2D link gets disconnected, no packets are received and the link quality cannot be estimated. To avoid trying to transmit data packets when a D2D link is disconnected, the implemented scheme logs the time elapsed from the last packet received from the D2D counterpart. If such time is higher than $T_{disconnection}$, the D2D data transmissions are also paused.

The link-aware opportunistic scheme has been implemented in a module at the MAC sub-layer. The module manages the data packets coming from the upper layers, and decides whether they are passed to the physical layer as soon as they are received from the upper layers, or they are stored in a buffer[3]. The decision to transmit or store the data packets is based on the experienced D2D link quality conditions. The implemented module also checks every $T_{Buffer}$ seconds whether the link quality conditions have improved and the stored data packets can be transmitted. An additional module has been implemented to log the RSSI and timestamp of the received packets from the D2D counterpart.

[3] The implementation of this module had to ensure that it would not interfere with processes from high layers that could affect the users' applications. In addition, it is important it does not modify any timing-dependent standard process such as the contention mechanisms to access the radio channel.

### B. Implementation at the Linux kernel

The link-aware opportunistic transmission scheme is implemented at the MAC sub-layer of the default 802.11 implementation in the Linux kernel (Ubuntu 13.04, 3.8.9-19 kernel version, <linux/net/mac80211>). The current implementation uses the Linux kernel Backports project that accelerates and facilitates modifications in the Linux kernel, and allows mounting the desired <mac80211> module changes without having to recompile the entire kernel. The 802.11 MAC changes when receiving packets are implemented in the <rx.c> file. These changes refer to the need to log the RSSI level and the timestamp of the received data, management and control packets from the counterpart. The changes have been implemented in the <ieee80211_rx()> function that is the first function executed when a packet arrives to the <mac80211> kernel module from lower layers. The transmission 802.11 MAC changes are implemented in the <tx.c> file. These changes refer to the implementation of the module that decides whether data packets are transmitted or stored in the buffer based on the experienced D2D link quality conditions. The module has been implemented in the <ieee80211_tx_frags()> function that is the last function executed in the <mac80211> module before 802.11 packets are sent to the lower layers of the protocol stack. The implemented module checks the Frame Control field to detect whether the packet to be transmitted is a data packet or a management or control packet. If the packet is a data packet, then the D2D link quality conditions are evaluated to decide whether the packet should be transmitted or stored in the buffer. If the link quality conditions are good enough, the data packet is sent to the lower layers using the <drv_tx()> function. If not, the data packet is stored in the buffer that has been implemented using the Work Queue (WQ) tool. WQ is an interruption handler that defines kernel deferrable functions that are executed at a configurable time (in this case $T_{Buffer}$) after the interruption is launched. A kernel deferrable function is then scheduled every time a data packet is stored in the buffer. When the $T_{Buffer}$ timer expires, the data packet is transmitted if the link quality conditions have improved[4]. If not, the packet is stored again in the buffer, and a new deferrable function is scheduled.

## IV. Experimental evaluation

The efficiency and performance of link-aware opportunistic D2D communications has been evaluated through a series of experiments where two portable devices (laptops) exchange a data file. The devices communicate using Ubiquiti SR71X cards (with AR9280 Atheros chipset) implementing IEEE 802.11g at 2.4GHz. The devices incorporate a packet sniffer to capture the 802.11 traffic. The sniffer was implemented using the open source <libpcap.h> library. This packet sniffer has been very valuable to monitor and analyze the D2D performance in the conducted field tests.

[4] It is possible that data packets arrive out of order at the receiver. This is the case because new data packets coming from upper layers could be processed before $T_{buffer}$ expires. If the link quality conditions have improved, these new data packets will be sent before the packets stored in the buffer. However, the packets can be re-ordered at the receiver.

### A. Configuration and evaluation scenario

A set of over-the-air measurements have been first conducted to tune the parameters of the implemented scheme (the complete tuning process cannot be here explained due to length restrictions). $T_{Buffer}$ has been set to 1s to consider the speed at which the link quality conditions can vary, but also the cost of evaluating such conditions. $T_{disconnection}$ has been set to 1.5s to detect disconnected D2D links ($T_{disconnection}$ needs to be higher than $T_{Buffer}$ to avoid confusing a pause with a disconnection). A first series of tests consisted in one of the devices walking away from the other one under LOS conditions. Different $RSSI_{thr}$ values were tested with the objective to identify the impact of this parameter on the D2D communications distance and performance (evaluated in terms of throughput and Packet Error Rate, PER). Following the conducted tests, $RSSI_{thr}$ was set to -74dBm since it allowed relatively large communication distances between D2D devices with good performance levels before the D2D transmissions were paused when the link quality dropped below -74dBm. The *Nb_Rx* and *Nb_BelowThr* parameters were set up with additional tests considering Non-LOS (NLOS) conditions between the two devices caused by the presence of dense vegetation or passing vehicles. Three different values were tested for these two parameters {3, 7, 12}. The opportunistic D2D performance improved with larger values of *Nb_Rx* and lower values of *Nb_BelowThr* under higher NLOS conditions (dense vegetation). This is the case because this setting allowed avoiding short D2D pauses under highly variable RSSI levels, and inefficient D2D communications under poor link quality conditions. When LOS conditions are only intermittently lost as a result of passing vehicles, better performance levels were obtained setting *Nb_Rx* to 3 and *Nb_BelowThr* to 7. With this setting, D2D transmissions were not paused when vehicles crossed but when vehicles stopped between the devices for a few seconds. The conducted tests showed that it is difficult to achieve an optimum configuration for all scenarios. As a result, a compromise solution was adopted and the *Nb_Rx* and *Nb_BelowThr* parameters were set to 7 and 3, respectively. The authors acknowledge that more tests could be conducted to further optimize the operation of the implemented scheme, and that context-aware adaptive settings could optimize further the performance (the cost would need to be considered though). However, the conducted experiments are considered sufficient for an adequate configuration of the implemented scheme in order to experimentally analyze the energy, capacity and QoS benefits of link-aware opportunistic D2D communications. It should also be noted that all opportunistic D2D configurations tested improved the performance compared to the default 802.11 implementation.

Different deployment scenarios have been evaluated but cannot be here reported due to length restrictions. In this paper, we focus on the evaluation scenario illustrated in Figure 1. The figure shows the initial location of the devices (*Node A* and *Node B*), their separation distance, and the route followed by the devices that move towards an intersection at pedestrian speed before turning around the corner. This scenario has been selected since it offers the possibility to test different length of time under which D2D links degrade by modifying the separation distance between devices. This paper reports the experimental tests conducted with separation distances between devices of 30m and 50m (such distances are approximately maintained during their route). At the beginning of each test, both devices are placed under LOS conditions to each other. An ad-hoc network is established between them, and a file transfer is launched using the UDPcast application. The nodes start moving and the LOS conditions are maintained until *Node A* turns around the corner. NLOS conditions are then experienced until *Node B* turns also around the corner. The considered distances between devices allow then studying the impact of different transmission times under NLOS conditions on the D2D performance. The tests finish when a 25Mbytes file is completely transmitted from *Node B* to *Node A*[5]. The selected file size ensures that the D2D transmissions finish after the two devices turn the intersection corner.

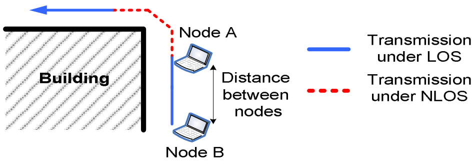
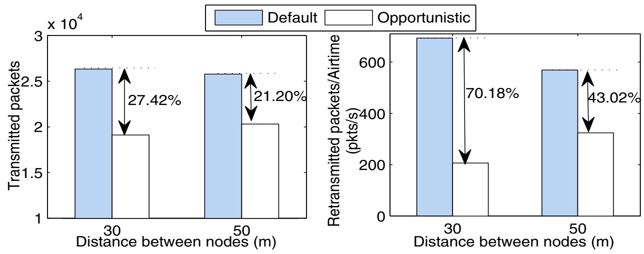



Figure 1. Evaluation environment.

## B. *Energy efficiency*

The implemented opportunistic transmission scheme has been designed to avoid unnecessary and inefficient D2D transmissions when the link quality degrades. Such degradation would result in high energy consumption levels since many packets can be received with error or transmitted with low-throughput transmission modes. This can be avoided by postponing the transmissions until better link quality conditions are experienced. The energy-efficiency benefits that can be achieved by exploiting opportunistic D2D communications are here analyzed by means of the total number of radio packets transmitted between Nodes *B* and *A* to transfer the 25Mbytes file. The total number of radio packets includes retransmissions. Figure 2.a shows the average total number of transmitted packets to transfer the 25Mbytes file. Results are presented for the default 802.11 Linux implementation (*Default*), and for the modified one that includes the designed link-aware opportunistic transmission scheme (*Opportunistic*). The results reported in Figure 2.a show that the *Opportunistic* approach can reduce the total number of transmitted packets by 27% and 21% compared to the *Default* one when the distance between the nodes is 30m and 50m, respectively. The reduction is smaller in the 50m case as the link quality degrades with the D2D distance and more robust transmission modes are used. The benefit obtained with the opportunistic scheme is due to a lower number of retransmitted packets since the 25Mbytes file is fragmented into 802.11 radio packets of approximately 1500bytes in the two configurations. This trend is observed in Figure 2.b that depicts the ratio of retransmitted packets per effective transmission time or airtime. The airtime represents the effective time that the radio channel is actually used during a D2D transmission (including the time needed for retransmissions). The airtime considers the time required for sending data packets that depends on the packet's length, the employed transmission mode, and the standard-based inter frame spaces and contention time (SIFS, DIFS, ACK time and congestion window) [3]. In this case, Figure 2.b shows that the differences between the *Opportunistic* and *Default* implementations increase to 70% and 43%, which clearly demonstrates how link-aware opportunistic schemes can significantly improve the energy-efficiency of D2D communications by avoiding inefficient transmissions.




a) Average total number of transmitted packets

b) Ratio of retransmitted packets per airtime

Figure 2. Energy efficiency.

## C. *Capacity*

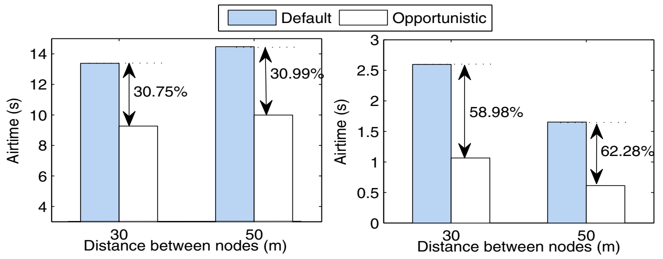
Reducing unnecessary or inefficient radio transmissions can also reduce the channel occupancy, and hence improve the capacity through spectrum reuse. The channel occupancy has been measured in the conducted field experiments by means of the airtime metric. Figure 3.a depicts the average airtime measured during the transmission of the complete 25Mbytes files with distances between the devices equal to 30m and 50m. The depicted results show that the use of opportunistic schemes in D2D communications can reduce the channel occupancy or airtime by approximately 30% when the distance between the devices is equal to 30m. When the distance increases, worse link quality conditions are experienced and more robust transmission modes are employed[6]. This results in an increase of the channel occupancy or airtime, although the benefits of the *Opportunistic* configuration with respect to the *Default* one are maintained. Figure 3.a shows that the use of the opportunistic scheme reduces the D2D airtime by scheduling the transmissions under good link quality conditions. It is important highlighting that this is done without increasing the end-to-end transmission time for the evaluated scenario. In fact, the *Opportunistic* configuration reduces the time needed to completely transmit the 25Mbytes file from *Node B* to *Node A*. While the *Default* implementation required 56.8 seconds to transmit the file when the distance between the nodes was equal to 30m, this value decreased to 55.8 seconds when using the implemented *Opportunistic* scheme. When the distance between nodes was equal to 50m, the *Default* implementation required 82.3 seconds to complete the transmission, and the *Opportunistic* implementation only 75.8 seconds.

The results shown in Figure 3.a correspond to the average airtime measured during the transmission of the complete

[5] With a distance of 30m, the tests last on average approx. 56 seconds with NLOS conditions experienced during approx. 21 seconds. These average values change to 80 and 35 seconds respectively with a distance of 50m.

[6] The *Opportunistic* scheme transmitted 34.3% of the packets using the 54Mbps data rate transmission mode when the distance was equal to 30m. This percentage was reduced to 14.6% with a distance of 50m.

25Mbytes file. On the other hand, Figure 3.b depicts the average airtime metric only in the area where the mobile devices experience NLOS conditions. The obtained results clearly show that the benefits of opportunistic schemes in D2D communications significantly increase when the link quality conditions degrade (NLOS conditions). In this case, the channel occupancy can be reduced by over 60%. It is also interesting noting that the airtime decreases when the distance between the devices increases to 50m. In the case of the *Default* implementation, this is due to the D2D disconnection suffered when the channel quality degrades. Such disconnection reduces the data being transmitted under NLOS, and therefore the measured airtime. When using the *Opportunistic* implementation, the link quality goes below $RSSI_{thr}$ at an earlier time when increasing the distance to 50m, which also reduces the data being transmitted and the airtime.



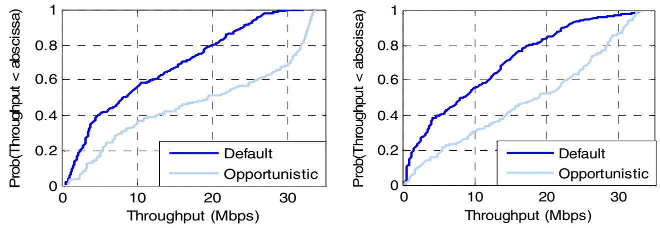

a) Average airtime for the D2D transmission of the complete file

b) Average airtime for the D2D transmission under NLOS conditions

Figure 3. Capacity.

### D. *Quality of Service*

The previous sections have experimentally demonstrated that link-aware opportunistic transmission schemes can improve the energy-efficiency and capacity of D2D communications by avoiding unnecessary and inefficient transmissions under bad link quality conditions. This operation can also improve the throughput experienced during D2D transmissions (and as a consequence reduce the end-to-end transmission time as shown in Section IV.C). The throughput is here computed as the ratio between the application data that is correctly received and the airtime necessary to correctly receive such data (including the time needed for retransmissions). The throughput experienced during D2D transmissions with the *Default* and *Opportunistic* configurations is shown in Figure 4 by means of a Cumulative Distributed Function (CDF). The obtained results show that when the distance between devices was equal to 30m, 50% of the D2D transmissions experienced a throughput equal or higher than 7.5Mbps with the *Default* 802.11 setting. This value increased by 166% when using the *Opportunistic* 802.11 implementation. Similar trends were observed when the distance between devices increases to 50m. The higher throughput levels experienced by the *Opportunistic* D2D implementation are obtained as a result of scheduling the transmissions under better link quality conditions which allows reducing the PER and using transmission modes with higher data rates. For example, the *Opportunistic* scheme transmits 68% of the packets using the 54Mbps and 48Mbps 802.11 transmission modes when the distance between devices is 30m. This percentage is reduced to 55% with the *Default* implementation. The average PER experienced during the D2D transmission of the 25Mbytes file was reduced from 53.48% when using the *Default* 802.11 implementation to 38.71% when considering the *Opportunistic* one.




a) 30m distance

b) 50m distance

Figure 4. Throughput CDF during D2D transmissions.

## V. CONCLUSIONS

This paper has presented the first experimental demonstration that link-aware opportunistic transmission schemes can help improve the efficiency, reliability and performance of D2D communications. In particular, the conducted study has shown that opportunistic schemes can significantly reduce energy consumption and channel occupancy (and thereby improve capacity), and improve the QoS of D2D communications. To conduct the study, the paper has presented a novel platform that modifies the Linux kernel to implement and test cross-layer schemes without degrading the performance achieved with commercial hardware. The implemented platform is released open source.

## ACKNOWLEDGMENT

This work is supported by the Spanish Ministry of Economy and Competitiveness and FEDER funds (TEC2011-26109), and the Government of Valencia (ACIF/2010/161).

## REFERENCES


[1] P. Phunchongharn, et al., "Resource allocation for device-to-device communications underlaying LTE-advanced networks", *IEEE Wireless Communications, vol. 20, no. 4, pp. 91-100*, Aug. 2013.

[2] B. Coll-Perales, et al, “Empirical Performance Models for P2P and Two Hops Multi-hop Cellular Networks with Mobile Relays”, *Proceedings of the 8th ACM PM2HW2N-2013 Workshop*, Nov. 2013.

[3] Y. Xiao and J. Rosdahl, “Throughput and Delay Limits of IEEE 802.11”, *IEEE Comms Letters, vol. 6, no. 8, pp. 355-357*, Aug. 2002.

[4] N. Zlatanov, R. Schober and P. Popovski, "Buffer-Aided Relaying with Adaptive Link Selection", *IEEE Journal on Selected Areas in Communications, vol. 31, no. 8, pp. 1530-1542*, Aug. 2013.

[5] M.I. Poulakis, et al., "Channel-Aware Opportunistic Transmission Scheduling for Energy-Efficient Wireless Links", *IEEE Transactions on Vehicular Technology, vol. 62, no. 1, pp. 192-204*, Jan. 2013.

[6] 3GPP TR 22.803. Technical Specification Group Services and System Aspects; Feasibility study for Proximity Services (ProSe). Mar. 2013.

[7] T. Korakis et al., “Cooperative Network Implementation Using Open-Source Platforms”, *IEEE Communications Magazine*, vol.47, no.2, pp. 134-141, 2009.

[8] K. Choumas, et al., “New prioritization schemes for QoS provisioning in 802.11 wireless networks”, *Proceedings of the 16th IEEE LANMAN Workshop*, pp 25-30, Sept. 2008.

[9] K. Gomez, et al., “On efficient airtime-based fair link scheduling in IEEE 802.11-based wireless networks”, *Proceedings of the 22nd IEEE* PIMRC, pp.930-934, Sept. 2011.

[10] M. Danieletto, et al., “On the Exploitation of the Android OS for the Design of a Wireless Mesh Network Testbed”, *Proceedings of the IEEE MILCOM*, pp. 1032-1038, Nov. 2013.