

Measuring transmission opportunities in 802.11 links

Domenico Giustiniano*, David Malone[†], Douglas J. Leith[†] and Konstantina Papagiannaki⁺

*Università degli Studi di Roma-Tor Vergata, Dipartimento di Ingegneria Elettronica; [†]Hamilton Institute, NUI Maynooth, Ireland; ⁺Intel Research, Pittsburgh.

Abstract— We propose a powerful MAC/PHY cross-layer approach to measuring 802.11 transmission opportunities in WLAN networks on a per-link basis. Our estimator can operate at a single station and it is able to i) classify losses caused by noise, collisions and hidden-nodes, and ii) distinguish between these losses and the unfairness caused by both exposed nodes and channel capture. Our estimator provides quantitative measures of the different causes of lost transmission opportunities, requiring only local measures at the 802.11 transmitter and no modification to the 802.11 protocol or in other stations. Our approach is suited to implementation on commodity hardware and we demonstrate our prototype implementation via experimental assessments. We finally show how our estimator can help the WLAN station to improve its local performance.

Index Terms— 802.11, CSMA/CA, channel measurement

I. INTRODUCTION

The practical performance of 802.11 depends on the availability of transmission opportunities at the underlying 802.11 card. In all the cases when transmission opportunities are lost, 802.11 stations will be affected by throughput drops, higher delay and unfairness. Lost transmission opportunities, arise from the *combined MAC and PHY* environment at both sender and receiver on each 802.11 link. Thus, their impact cannot be estimated purely on the basis of PHY measurements (see e.g. [1]–[5]), such as signal-to-noise ratio (SNR) at the receiver. Alternative high-level characterizations, such as throughput and delay statistics, can only determine the effect but not the *cause* of a lost transmission opportunity. In fact, the nature and number of transmission opportunities are a combination of MAC and PHY layer effects.

The need for a *cross-layer technique* can be readily seen, for example, from the fact that hidden nodes effects, exposed nodes, capture effects and so on are all associated with cross-layer issues. Other effects are instead due only to features of the MAC layer, such as the collisions, or to PHY layer, such as a low signal-to-noise ratio. Furthermore, tasks such as rate adaptation, contention window selection, power control and carrier sense selection — all essential for improving and optimizing the network performance — are under the control of the *transmitter*.

Despite the difficulty of measuring and understanding transmission conditions, the potential benefits arising from the availability of accurate and reliable data are considerable and depend crucially on the availability of suitable measurements

of the transmission quality at each link. It is the current lack of such measurements that underlies the poor performance of many approaches currently implemented in commodity hardware. For example, at present rate adaptation is commonly based on the number of transmission retries (e.g. a typical approach might involve lowering the rate after n retries and increasing the rate after m successful transmissions). However, since the number of retries is affected not just by channel noise but is also closely linked to the number of contending stations (with associated collision related losses), this can easily lead to poor performance [6]. Analogous problems occur in the presence of hidden nodes, e.g. see [7]. The availability of a measure of the loss rate specifically induced by channel noise would potentially allow much more effective rate adaptation algorithms to be employed. Similarly, channel selection algorithms are fundamentally related to channel impairments and typically depend upon the availability of an appropriate channel quality metric, which can then be optimized by a suitable search over available channels. Effective carrier sense adjustment is also strongly dependent on link measurements. Causes of lost transmission opportunities and corresponding remedies available at the transmitter are shown in Table I.

Measurement of transmission opportunities is particularly topical since the trend towards increasingly dense wireless deployments is creating a real need for effective approaches for channel allocation/hopping, power control, interference mitigation [8], [9], etc. New applications, such as mesh networks and in-home content distribution, are creating new quality of service demands that require more sophisticated approaches to radio resource allocation [10].

In this paper we propose a powerful new transmitter-side MAC/PHY cross-layer approach to measuring the transmission opportunities in 802.11 WLANs. Unlike previous approaches, we explicitly classify lost transmission opportunities into noise-related losses, collision induced losses, hidden-

Transmission Impairment	Remedy by adjusting
Collision	Contention window
Noise	PHY Rate/Channel
Exposed Node	Carrier Sense/Channel
Hidden Node	Carrier Sense/Channel
Capture Effect	Transmission Power

TABLE I
TRANSMISSION IMPAIRMENTS AND AVAILABLE REMEDIES AT
TRANSMITTER.

We gratefully acknowledge the help of Richard Gass at Intel. Supported by Science Foundation Ireland grants IN3/03/I346 and 07/IN.1/I901.

node losses and distinguish these losses from the unfairness caused by exposed nodes and capture effects. Our approach distinguishes among these different types of impairments on a per-link basis, and provides separate quantitative measures of the severity of each. We thus make available new measures that we expect to be of direct use for rate adaptation, channel allocation, *etc.* We take advantage of the native characteristics of the 802.11 protocol (such as timing constraints, channel busy detection and so on), without requiring any modification to the 802.11 standard. Our approach is suited to implementation on commodity hardware and we demonstrate both a prototype implementation and experimental measurements. It is vital to demonstrate operation in a real radio environment, not only because of the difficulty of developing realistic radio propagation models, but also because important impairments such as hidden-nodes and capture effects are affected by low-level issues (e.g. interactions between amplifier and antenna design as well as radio propagation) that are difficult to model in simulations. We note that many of the measurements presented are new and of interest in their own right.

The paper is organized as follows. In section II we review related work and show how our estimator differs from the state of the art. In section III we briefly review the 802.11 MAC and then categorize the main causes of lost transmission opportunities. In sections IV and V we introduce our estimation approach. We describe our testbed setup in section VI and present extensive experimental measurements in both section VII and section VIII, evaluating this approach in a wide range of real radio environments. Finally we summarize our conclusions in section IX.

II. RELATED WORK

Previous work on estimating 802.11 channel conditions can be classified into three categories. First, *PHY link-level* approaches use SNR and Bit-Error Rate (BER). Second, *MAC approaches* rely on throughput and delay statistics, or frame loss statistics derived from transmitted frames which are not ACKed and/or from signaling messages. Finally *cross-layer MAC/PHY approaches* aim to combine information at both MAC and PHY layers.

Most work on PHY layer approaches is based on SNR measurements, e.g. [2], [5]. The basic idea is to a-priori map SNR measures into link quality estimates. However, there are limitations of this approach. First, frames are usually lost at the receiver (802.11 data packet) and not at the transmitter (802.11 ACK packet). Consequently, SNR measurements at the receiver side must be passed to the transmitter, which is ‘responsible’ for improving the link performance via carrier-sense, rate, transmission power and channel adaptations. This requires the use of a modified 802.11 protocol to send feedbacks to the transmitter. Second, the mapping between measured SNR and delivery probability rate is generally specific to each link [1] and may be time-varying, thus requiring a continuous update of the statistics from the receiver to the transmitter. Finally, the correlation between SNR and actual packet delivery rate can be weak in some situations [3].

To by-pass problems related to SNR-only measurements, [16] performs loss diagnosis by examining the error pattern

within a physical-layer symbol in order to expose statistical differences between collision and weak signal based losses. In the proposed system, the AP relays the entire packet, received in error, back to the client for analysis, with a consequent overhead for the network. Furthermore the analysis is limited to collision and noise related losses.

In contrast, in this paper we provide a local methodology to explicitly classify noise-related losses, collision induced losses, hidden-node losses and impairments caused by exposed nodes and capture effects. As part of our validation, we compare our scheme to a receiver-side estimator for the loss diagnosis. This receiver-based scheme can not only differentiate between collision and weak signal based losses, but in general, among noise, collisions and hidden-node losses.

Regarding MAC approaches, RTS/CTS signals can be used to distinguish collisions from channel noise losses, e.g. [22], [23]. However, such approaches can perform poorly in the presence of hidden nodes as we demonstrate in section VII. [24] considers an approximate MAC layer approach for detecting the presence of hidden nodes but does not consider other types of transmission impairments.

With regard to combined MAC/PHY approaches, early work related to the present paper is presented in [18], [19]. However, this uses a channel busy/idle approach that is confined to distinguishing between collision and noise related losses and does not allow consideration of hidden nodes or exposed node and capture effects. In [20] techniques to separate noise from collision losses are considered, but hidden node techniques are left as future work. In [21] the importance of root-cause analysis of transmission failures is considered and a monitoring system based on multiple sniffers is designed.

III. CSMA/CA PROTOCOL AND RELATED IMPAIRMENTS

A. CSMA/CA protocol

In 802.11 WLANs, the basic mechanism controlling medium access is the *Distributed Coordination Function* (DCF) [11]. This is a random access scheme, based on Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA). In the DCF Basic Access mode, a station with a new packet to transmit selects a random backoff counter. Time is slotted and if the channel is sensed idle the station first waits for a Distributed InterFrame Space (DIFS), then decrements the backoff counter each PHY time slot. If the channel is detected busy, the countdown is halted and only resumed after the channel is detected idle again for a DIFS. Channel idle/busy status is sensed via:

- CCA (Clear Channel Assessment) at the physical level which is based on a carrier sense threshold for energy detection, e.g. -80dBm . CCA is expected to be updated every physical slot time. It aims to detect transmissions within the interference range.
- NAV (Network Allocation Vector) timer at MAC level which is encapsulated in the MAC header of each 802.11 frame and is used to predict the end of a received frame on air. It is naturally updated once per packet and can only gather information from stations within the decoding range. This method is also called virtual carrier sense.

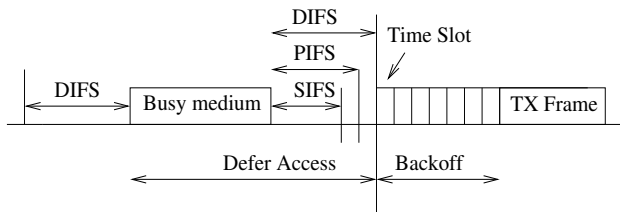


Fig. 1. DCF protocol summary.

The channel is detected as idle if the CCA detects the channel as idle and the NAV is zero. Otherwise, the channel is detected as busy. A station transmits when the backoff counter reaches zero. The countdown process is illustrated schematically in Fig. 1. The 802.11 handshake imposes a half-duplex process whereby an acknowledgment (ACK) is always sent by the receiver upon the successful receipt of a unicast frame. The ACK is sent after a period of time called the Short InterFrame Space (SIFS). As the SIFS is shorter than a DIFS, no other station is able to detect the channel idle for a DIFS until the end of the ACK transmission.

In addition to the foregoing Basic Access mode, an optional four way handshaking technique, known as Request-To-Send/Clear-To-Send (RTS/CTS) mode is available. Before transmitting a packet, a station operating in RTS/CTS mode reserves the channel by sending a special Request-To-Send short frame. The destination station acknowledges the receipt of an RTS by sending back a CTS frame, after which normal packet transmission and ACK response occurs.

The DCF allows the fragmentation of packets into smaller units. Each fragment is sent as an ordinary 802.11 frame, which the sender expects to be ACKed. However, the fragments may be sent as a burst. That is, the first fragment contends for medium access as usual. When the first fragment is successfully sent, subsequent fragments are sent after a SIFS, so no collisions are possible. In addition, the medium is reserved using virtual carrier sense for the next fragment both at the sender (by setting the 802.11 NAV field in the fragment) and at the receiver (by updating the NAV in the ACK). Burst transmission is halted after the last fragment has been sent or when loss is detected.

B. Losses of Transmission Opportunities

In this section we categorize the main impairments that can affect transmissions between an 802.11 sender and receiver. Before proceeding, it is important to emphasize that a two-way (or four-way with RTS-CTS) handshake is used in 802.11. Hence, the quality of a link is determined by conditions at both the sender and the receiver stations. For example, low link-quality at the receiver can mean that data packets transmitted by the sender cannot be decoded at the receiver. Similarly, low link-quality at the sender can mean that ACK packets transmitted by the receiver cannot be decoded at the sender. It follows immediately that:

- Measuring the SNR (or other local properties) at either the sender or receiver alone is insufficient to determine its quality. Instead it is necessary to recognize the intrinsically two-way nature of each link.

- Links are directional as data packets and ACKs may have different properties e.g. coding rate, duration, NAV protection. Collisions and interference with transmissions by other stations can affect each end of a link differently.
- Since each station is typically located in a different physical position, its local radio environment is generally different from that of other stations. Hence we need to measure the transmission opportunities between each sender-receiver pair individually. In particular, we cannot reliably infer the properties of one link from measurements taken on another link, even if the links share a common sender e.g. the AP in an infrastructure mode WLAN. Further, due to the directional nature of link quality (see above) we need to measure quality in each direction separately and generally cannot use measurements from one direction to reliably infer the quality in the opposite direction. An example illustrating this is shown later in the paper, see section VIII-B.

As we will see, the manner in which link impairments are manifested is closely linked to the interaction between MAC and PHY operation. We distinguish five main types of link impairment when using the 802.11 DCF.

1) *Collisions*: Collisions are part of the correct operation of CSMA/CA. Collisions occur when two or more stations have simultaneously decremented their backoff counter to 0 and then transmit. Note that collisions can only occur on data packet transmissions. The level of collision induced packet losses is strongly load dependent. For example, 802.11b with four saturated nodes has a collision probability of around 14%, while 20 saturated have a collision probability of around 40% (numbers from the model in [15]). We denote by p_c the probability that a transmitted data frame is lost due to a collision.

2) *Hidden nodes*: Frame corruption due to concurrent transmissions other than collisions are referred to as hidden node interference. We denote by $p_{h,data}$ the probability that a data transmission fails to be received correctly due to hidden node interference. Similarly, we denote by $p_{h,ack}$ the probability that an ACK transmission is lost due to hidden node interference. A lost data packet or a lost ACK both lead to a failed transmission and so we combine data and ACK losses into an overall hidden node error probability p_h .

3) *Noise errors*: Frame corruption due to sources other than transmissions by other 802.11 stations are referred to as noise losses. We denote by $p_{n,data}$ (respectively, $p_{n,ack}$) the probability that a data (respectively, ACK) frame is lost due to noise related errors. Since data and ACK losses both lead to a failed transmission we lump these together into a combined noise loss probability p_n .

4) *Exposed nodes*: The inability to transmit 802.11 frames is not just due to link losses. In particular, the carrier sense mechanism used in 802.11 to sense channel busy conditions may incorrectly classify the conditions. We denote by p_{exp} the probability that a slot is erroneously detected as busy when in fact a successful transmission could have been made. Such errors lead to an unnecessary pause in the backoff countdown and so to a reduction in achievable throughput.

5) *Capture effect*: A second impairment which does not cause losses is the so-called physical layer capture (PLC). Specifically, we denote by p_{plc} the probability of successful reception of a frame when a collision occurs. This can occur, for example, when the colliding transmissions have different received signal power — the receiver may then be able to decode the higher power frame. For example [17] shows that for 802.11b PLC can occur when a frame with higher received power arrives within the physical layer preamble of a lower power frame. Our measurements — not shown here for lack of space — have confirmed this finding and found similar behavior for 802.11g. Differences in received power can easily occur due to differences in the physical location of the transmitters (one station may be closer to the receiver than others), differences in antenna gain etc. The physical layer capture effect can lead to severe imbalance of the network resource and hence in the throughputs achieved by contending stations (and so to unfairness).

IV. ESTIMATING LOSSES OF TRANSMISSION OPPORTUNITIES

Our aim is to develop a sender-side estimator capable of distinguishing the different types of impairments and providing quantitative measurements of the related losses. To do this we make use of the key observation that these impairments are intimately related to MAC operation. We therefore exploit the flexibility already present in the 802.11 MAC to enable us to distinguish the impact of the different impairments. The technique can use live traffic, and so can naturally estimate each active link being used at a station. If measurements of inactive links are required, this could be achieved by using probe traffic.

We make use of the following properties of the 802.11 MAC:

- Time is slotted, with well-defined boundaries at which frame transmissions by a station are permitted.
- The standard data-ACK handshake is affected by all types of link impairment considered and a sender-side analysis can reveal any loss.
- When fragmentation is enabled, second and subsequent fragment transmissions are protected from collisions and hidden nodes by the NAV values in the fragments and ACKs. We treat hidden nodes that are unable to decode either NAV value as channel noise. Instead of using fragments, we could use TXOP packet bursting, although this is only available in 802.11e [12], and would require the NAV value set in the MAC ACK. RTS/CTS might also be used, but in practice can perform poorly, as we will demonstrate later in the paper.
- Transmissions occurring before a DIFS are protected from collisions. This is used, for example, to protect ACK transmissions, which are transmitted after a SIFS interval. The 802.11 DCF also permits transmissions after a PIFS interval (with $SIFS < PIFS < DIFS$). While the full 802.11 Point Coordination Function (PCF) is rarely fully implemented, the ability to transmit after a PIFS is widely available; if not as part of the PCF, it is available as part

WME (wireless multi-media extensions), the subset of 802.11e, commonly available on commodity hardware.

In the following sections we consider in more detail how these properties can be exploited to obtain new powerful measurements of the quality of each link.

A. Estimating Noise Errors

Consider a station sending fragmented packets to a given receiver. Each fragment is immediately acked by the receiver when it arrives, allowing detection of loss. Fragments are sent back to back with a SIFS interval between them. Hence, second and subsequent packets are protected from collisions. Importantly, fragment ACK frames update the NAV and so the fragment-ACK handshake is akin to an RTS-CTS exchange from the point of view of hidden nodes¹. Hence, second and subsequent fragments are also protected from hidden node collisions. That is, while the first fragment will be subject to collisions, noise and hidden node errors, subsequent fragments are only subject to noise errors and we have that

$$\mathbb{P}[\text{fragment success}] = A_S/T_S = (1 - p_n), \quad (1)$$

where the station transmits T_S second and subsequent data frames and of these A_S are successful because an ACK is received. We can therefore directly estimate the probability of noise errors p_n from the fraction of second and subsequent fragments with no ACK,

$$p_n = 1 - A_S/T_S \quad (2)$$

Since the impact of noise losses may depend on the frame length (longer frames typically having higher probability of experiencing bit errors), we equalize the length of the fragments we send and transmit fragments of length equal to the packet size used for regular data transmissions. The frame loss rate estimated from fragment measurements can then be reliably applied to estimate the loss rate for other transmissions.

B. Estimating Hidden Node Interference

We now distinguish frame losses due to hidden node interference. To achieve this we exploit the fact that frames transmitted after a PIFS are protected from collisions since other transmissions must defer for a DIFS interval after sensing the channel to be idle, with $DIFS > PIFS$. Losses on PIFS frames are due either to noise or hidden node interference. That is,

$$\mathbb{P}[\text{PIFS success}] = A_1/T_1 = (1 - p_h)(1 - p_n), \quad (3)$$

where the station transmits T_1 data frames after a PIFS and of these A_1 are successful because an ACK is received. We can now use our estimate of p_n (based on fragment loss measurements, see equation (2)), to allow estimation of the probability p_h of hidden node losses as:

$$p_h = 1 - (A_1 \cdot T_S)/(A_S \cdot T_1). \quad (4)$$

¹As mentioned, we do not rely on RTS/CTS since it can perform poorly, see Section VII.

C. Estimating Collision Rate

Consider a station sending ordinary data packets (i.e. sent after DIFS and not fragmented) to a given receiver. Suppose that over some time period the station contends and transmits data frames T_0 times and of these A_0 are successful because an ACK is received. As discussed previously, the possible sources of frame loss are: collisions, hidden nodes and noise errors. Assuming that these sources of frame loss are independent, if the station transmits the probability of success over the link is:

$$\mathbb{P}[\text{success}] = A_0/T_0 = (1 - p_c)(1 - p_h)(1 - p_n). \quad (5)$$

Finally p_c can be estimated from Eq. (5) and (3):

$$p_c = 1 - (T_1 \cdot A_0)/(T_0 \cdot A_1). \quad (6)$$

V. IMPAIRMENTS TO TRANSMISSION OPPORTUNITIES THAT DO NOT LEAD TO FRAME LOSS

Section IV presents a straightforward approach for estimating the magnitude of link impairments that lead to frame loss, namely collisions, hidden nodes and noise. The estimates require only very simple measurements that are readily available on commodity hardware. In this section we now consider methods for estimating capture and exposed node effects. These impairments do not lead directly to frame losses, but can nevertheless lead to unfairness in throughput/delay.

In order to estimate capture and exposed node effects we make use of additional measurements. In particular, measurements of channel idle and busy periods. Here idle/busy refers to time as measured in MAC slots rather than in PHY slots. In the next section we discuss MAC slots in more detail. Then we discuss estimating capture and exposed node effects. Note that while these additional measurements offer further insight into the wireless environment, they are not necessary to estimate the basic quantities p_c , p_n and p_h .

A. MAC slots

The slotted CSMA/CA process creates well-defined boundaries at which frame transmissions by a station are permitted. The time between these boundaries we call MAC slots (as distinct from PHY slots). Considering operation from the viewpoint of a station, say station 1, we have the following possibilities:

- 1) Station 1 has transmitted and received an ACK. We call these slots *successful transmissions*.
- 2) Station 1 has transmitted, timed-out while waiting for an ACK and is about to resume its backoff. We call these slots *unsuccessful transmissions*.
- 3) Station 1 has seen the medium as idle and, if backoff is in progress, has decremented its backoff counter. We call these *idle slots*.
- 4) Station 1 has detected the medium as busy due to one or more other nodes transmitting, and has suspended its backoff until backoff can resume. We call these slots *other transmissions*, and include both successful and unsuccessful transmissions of other stations. Note that

each busy period is counted as a single slot, so these busy slots are closer to the MAC's view than the PHY's.

These events are illustrated (not to scale) in Fig. 3. Transmissions by station 1 are only permitted at event boundaries.

We also make the following assumptions:

Assumption 1. The probability that at least one other station transmits in an arbitrary slot does not depend on whether station 1 transmits or not.

Assumption 2. The collision probability is independent of the backoff stage of station 1.

With these assumptions, the probability of a collision is then precisely the probability that at a slot boundary the channel is busy due to a transmission by one or more other stations. We note that our assumptions are reasonable in a distributed random access MAC scheme such as CSMA/CA and, indeed, these assumptions are central to well-established models of 802.11 operation such as that of Bianchi [15] and others (e.g. the nonsaturated heterogeneous model in [27]).

B. Capture and Exposed Nodes

Suppose there are R MAC slots in which our station does not transmit and that I of these are *idle slots* and $R - I$ are *other transmissions* slots. These quantities can be measured by appropriate sensing of the channel idle/busy status (using both CCA and NAV). Under ideal conditions, the ratio $R - I/R$ is a measure of the probability of collision [18], [19]. However, this carrier sense-based measurement is biased by impairments that do not generate packet losses. These are the slots where either the channel is busy when, in fact, a transmission would be successful (exposed nodes) or the slots where we expect a collision loss but we do not collide (capture effect). We therefore have that,

$$p_c + p_{exp} + p_{plc} = \frac{R - I}{R}, \quad (7)$$

where p_c is the collision probability, p_{exp} the probability that the channel is sensed busy due to exposed node behavior and p_{plc} the probability that the channel is sensed busy due to capture effects. Combining our estimate of p_c from eq. (6) with the additional information in (7), we can estimate:

$$p_{exp} + p_{plc} = (T_1 \cdot A_0)/(T_0 \cdot A_1) - I/R. \quad (8)$$

In effect we are estimating the number of collisions losses that we expect based on the carrier sense environment and comparing it with the actual collision rate. The discrepancy, if any, provides a measure of exposed node and capture effects — both of which are associated with apparently busy slots during which a successful transmission can in fact take place.

Note that these idle/busy measurements can also be used to estimate the collision probability in the absence of exposed node or capture effects (see [18], [19]) but this is not possible in the more general setting considered here.

VI. IMPLEMENTATION ON COMMODITY HARDWARE AND TESTBED SETUP

A. Implementation

We have implemented the foregoing estimators using a combination of driver and firmware modifications to commodity

Counter	TX counters	slot counters	Requires
T_0	TX of normal traffic		Normal DCF TX
T_1	TX of PIFS traffic, first frag.		After PIFS TX
T_S	TX of subsequent frag.		Fragmented TX
A_0	ACK of normal traffic		Normal DCF TX
A_1	ACK of PIFS traffic, first frag.		After PIFS TX
A_S	ACK of subsequent frag.		Fragmented TX
I		idle slots	idle counters
R		slots we do not TX in	busy counters
Symbol	Probability of	Estimator	Notes
p_c	collision	$1 - (T_1 \cdot A_0)/(T_0 \cdot A_1)$	clamp to [0,1]
p_n	noise interference err.	$1 - A_S/T_S$	
p_h	hidden node err.	$1 - (A_1 \cdot T_S)/(A_S \cdot T_1)$	clamp to [0,1]
$p_{exp} + p_{plc}$	exposed and capture effect	$(T_1 \cdot A_0)/(T_0 \cdot A_1) - I/R$	clamp to [0,1]

Fig. 2. Summary of measurements used and proposed estimators.

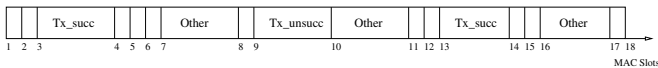


Fig. 3. MAC slot boundaries at which transmissions are permitted. Different types of MAC slot are possible: idle slots (corresponding to PHY slots), busy slots due to transmissions by other stations (marked “Other”) and busy slots due to transmissions the station of interest (marked “Tx.”). “Other” transmissions include both successful and unsuccessful transmissions.

network cards using the Atheros AR5212/AR5213 and Intel 2915ABG chipsets. The proposed estimators are summarised in Table 2. The TX counters would be maintained per-link (i.e., for each active physical destination). In the case of a station in a traditional infrastructure mode network, all traffic is sent via the AP, so only one set of counters is required.

The estimators of collision rate, hidden node and noise errors described in Section IV can be implemented via straightforward driver modifications. In our work they have been mainly tested on Atheros cards and the widely used MADWiFi driver. To transmit frames after a PIFS interval we made use of the WME (Wireless Multimedia Enhancements) features, which allow dynamic adjustment of the TXOP, CWmin and AIFS parameters for each Access Category of 802.11e. In particular, we created an access category with MAC settings $CWMin=CWMax=AIFSN=TXOP=0$. All traffic sent via the queue associated with this access category is then transmitted using PIFS. A second access category and queue is defined for normal traffic. On this queue, data packets are fragmented in two fragments, which is sufficient for assessing our estimator.² By appropriately directing packets to these two queues we can collect statistics for the overall number of transmissions T_0 , T_1 and T_S and number of successful transmissions A_0 , A_1 and A_S (transmissions for which a MAC ACK is received). In our implementation packets are allocated between queues at the driver level, although other solutions are possible.

The estimators in section V require measurement of the number of R and I busy and idle MAC slots. This requires carrier sense information. We modified the card firmware and microcode on cards using the Intel 2915ABG chipset to perform the necessary measurements and to expose these to the driver. Recent versions of Atheros driver provide access to

²Other traffic configurations are possible, e.g. to fragment only the PIFS traffic.

hardware registers that allow us to infer the number of idle and busy slots (e.g. see [25]), however we have conducted these tests with the Intel chipset. Our implementation implicitly uses the same carrier-sense threshold as the rest of the MAC.

We will also cross-validate a number of our results based on the number of CRC errors, CRC_{err} , observed at a receiving STA. This counter has been also retrieved from the microcode in Intel cards, and driver code in Atheros cards. This cross-validation is described in detail in section VI-C.

B. Testbed setup

To evaluate the estimators we performed experimental measurements over a wide range of network conditions, of which we present a subset here. Our testbed consists of Soekris net4801 devices running Linux and configured in infrastructure mode. Stations transmit 1400 byte UDP packets to an AP equipped with a NIC using the Intel 2915ABG chipset or Atheros AR5213 chipset, according to the specific test. As the impact of impairments can be high in even not-saturated conditions [27], we have considered a range of traffic loads. When not explicitly stated, nodes can assumed to be saturated. Unless otherwise specified, the physical rate is set to 6Mbps in each station, time slots are set to $20\mu s$ on both Intel and Atheros NICs and the carrier sense threshold for the Intel NICs was set to $-80dBm$, while the carrier sense level used with the Atheros NICs is the default value (set in the binary component — HAL — of the Atheros MADWiFi driver). In all experiments, automatic rate selection and the RTS/CTS mechanism are disabled unless otherwise stated. Antenna diversity functionality is also disabled (see [4] and therein), together with any proprietary mechanisms at MAC level. External interference levels are measured using a spectrum analyzer. Link impairments are generated as follows:

- *Noise errors* In the testbed we modify the SNR of a link by a combination of adjusting the physical separation of stations and/or adjustment of the transmit power used. In this way we can roughly control conditions to allow investigation of the ability of the proposed estimator to measure the level of frame losses due to noise errors on a link.
- *Collisions* The level of collision induced losses is adjusted by varying the number of contending stations and their offered traffic load.

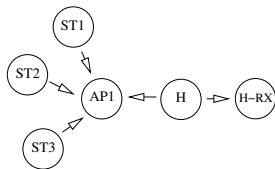


Fig. 4. Topology for hidden node tests.

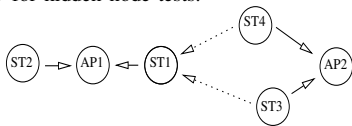


Fig. 5. Topology for exposed node tests.

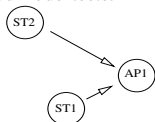


Fig. 6. Topology for physical layer capture tests.

- *Hidden nodes* Hidden node effects are evaluated using scenarios based on the setup illustrated in Fig. 4. We have a number of transmitting nodes and a receiver. The hidden node transmits to an independent receiver. We ensure that the following conditions hold: the link from the transmitter to our receiver is of high quality in isolation; the link from the hidden node to the hidden receiver is of high quality in isolation; a link can *not* be established from the transmitter to the hidden node; losses occur when the hidden node operates at the same time as the transmitter.
- *Exposed nodes* Exposed nodes are investigated via a setup with up to two interfering WLANs, as depicted in Fig. 5. In more detail, $ST1$ and $ST2$ are associated to $AP1$ (WLAN 1), while $ST3$ and $ST4$ are associated to $AP2$ (WLAN 2). In WLAN 1 we verify that i) $ST1$ receives the signals from WLAN 2 ($ST3$ and $ST4$) at higher strength than the carrier sense threshold ii) the $ST1 \rightarrow AP1$ link³ is of higher signal quality than the $ST3 \rightarrow AP1$ and $ST4 \rightarrow AP1$ links, so that $AP1$ may successfully decode any signal from $ST1$, despite the interference from WLAN 2.
- *Capture effects* Capture effects are studied using the setup illustrated in Fig. 6. Two stations $ST1$ and $ST2$ are associated to $AP1$. We verify that the $ST1 \rightarrow AP1$ link is of higher signal quality than the $ST2 \rightarrow AP1$ link such that transmissions by $ST1$ are successfully received at $AP1$ even when they collide with transmissions by $ST2$ i.e. $ST1$ can capture the channel.

C. Cross-Validation of Frame Loss Impairments

To help validate the sender-side measurements obtained using the estimator in the previous section, in our experimental tests we also perform loss diagnosis at the receiver-side. Particularly we make use of the following independent measurements for the differentiation of the noise-related losses, collision induced losses and hidden-node losses.

³We denote by $A \rightarrow B$ a link with data sent from A to B.

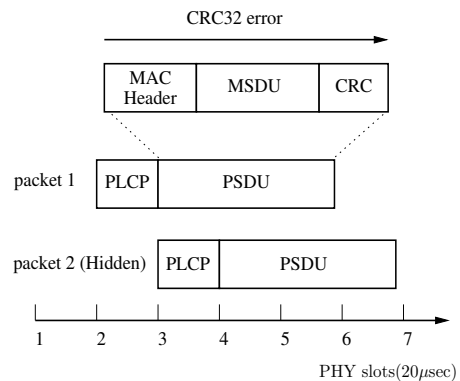


Fig. 7. Hidden node errors for an 802.11 frame (not to scale).

The 802.11 frame consists of a PLCP (Physical Layer Convergence Preamble) and MAC payload called the PSDU (Physical Service Data Unit). Each PSDU is protected with a 32 bit Cyclic Redundancy Check (CRC checksum). At the PHY level, errors in frame reception can be classified as either PHY or CRC errors:

- an error occurs on the PLCP preamble or header. We call these PHY errors.
- the PLCP is correctly decoded but the PSDU CRC fails: we call this a CRC32 error. Note that the presence of a CRC32 error notification on a received frame implies that no errors occurred in the PLCP.

We analyze the count of CRC32 errors for validation measurements, that is we consider when collisions, channel noise and/or hidden nodes result in CRC errors:

- 1) *Collisions* First, note that in a collision two or more transmit stations have chosen the same PHY slot to start transmission. We assume that a receiver station will not only observe this as a *busy* slot, but that it will also detect either a PHY error or, in the case of physical layer capture in the PLCP, a CRC error. We split the probability of collision,

$$p_c = p_{c1} + p_{c2}, \quad (9)$$

where p_{c1} is the probability of a collision resulting in a PHY error and p_{c2} the probability of a collision resulting in a CRC error. Thus p_{c2} collisions will be observed by the CRC estimator.

- 2) *Noise errors* Second, consider channel noise. Typically the PLCP is sent at a substantially lower rate than the PSDU, so we assume that channel noise never results in a PHY error, but instead in a CRC error.
- 3) *Hidden nodes* Finally, consider the impact of hidden nodes. The receiver will see a certain number of hidden node errors as simple collisions, when a hidden node and an ordinary node select the same slot, as illustrated at point 1 in Fig. 7. These will contribute to p_c . However, hidden-node transmissions beginning in later slots (i.e., after an ordinary node has already started) may result in more complex errors. In our experiments we use 802.11g transmissions with a PLCP of $20\mu s$ and the 802.11b compatible slot length of $20\mu s$. For this setup, shown in

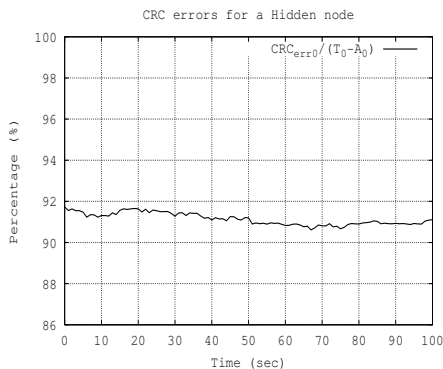


Fig. 8. Probability of CRC errors with a hidden node.

Fig. 7, we expect all of the hidden node errors that are not simple collisions to result in CRC errors, because the hidden node will not transmit until after the PLCP has been transmitted.

To confirm this assumption, we took 2 hidden nodes transmitting at 300fps (frames per second). Fig. 8 shows the fraction of retry errors at the transmitter that are mapped into CRCerr frames at the receiver. We see a consistent level of about 91%. The remaining 9% are attributed to both nodes choosing to transmit in the same slot thus leading to PHY errors, as we expect.

Thus, the CRC errors seen at the receiver satisfy:

$$\frac{CRCerr}{R - I} = p_n + p_h + p_{c2} - (p_n + p_h)p_{c1} - (p_n + p_h)p_{c2} \approx p_n + p_h + p_{c2}$$

where $CRCerr$ is the number of CRC32 errors and $R - I$ is the number of busy MAC slots seen at the receiver.

VII. EXPERIMENTAL ASSESSMENT

In this section we present experimental measurements to validate and explore the practical utility of the proposed estimators. We argue that experimental testing is vital when assessing link quality estimators since issues such as complex radio propagation effects, real antenna behavior, front-end amplifier issues *etc* can all have an important impact on performance yet are difficult to capture accurately in simulations. Experimental testing also highlights implementation issues, demonstrates the practicality of operation on commodity hardware, and generally helps to build greater confidence in the viability of the proposed approach. Our results are presented as the in terms of quantities such as p_c , p_h , and p_n . Throughputs figures are not presented due to space constraints, but can be inferred from the fraction of packets not subject to some impairment.

A. Collisions only, no noise, no hidden nodes

We begin by considering a simple scenario with a clean channel and no hidden nodes. A low level of RF interference is confirmed by a spectrum analyzer. We vary the number of contending wireless stations so as to vary the collision rate. Each station generates traffic at a rate of 300fps, which is

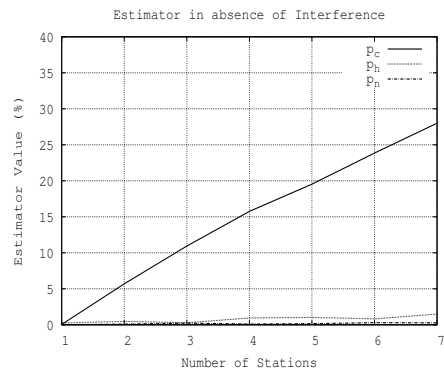


Fig. 9. Estimates of p_c , p_h , and p_n vs. number of contending stations. Clean channel, no hidden nodes.

sufficient to saturate the network, for an interval of 600s. 10% of the traffic is sent through the PIFS queue, while the rest is sent through the BE queue.

Fig. 9 shows the measured estimates of p_c , p_h , and p_n , averaged over the experiment. We can immediately make a number of observations:

- The collision probability p_c increases with the number of stations, as expected.
- The noise loss probability p_n , estimated from measurements on subsequent fragments, is negligible, as expected.
- The hidden node loss probability p_h is consistently low, as expected.

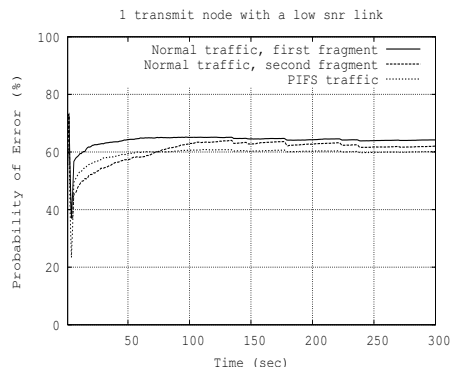
Although a simple test scenario, it is nevertheless encouraging that these initial tests indicate correct operation of the estimators. In particular, the ability to distinguish collision losses from noise and hidden node effects. We confirm this in the following sections by varying the level of noise and hidden node losses over a wide range of operating conditions.

B. Channel noise only, no collisions, no hidden nodes

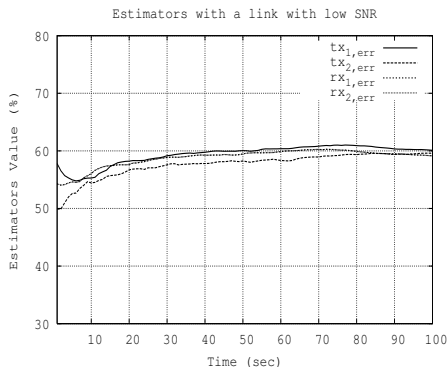
To explore the impact of channel noise, we begin this section by considering a setup with one transmitting and one receiving station and thus no collisions or hidden nodes (more complex setups with noise, collisions and hidden nodes are considered in later sections). The physical rate for transmission is fixed to 12Mbps and sending rate at 300fps, which saturates the transmit queue. The link is adjusted to have low SNR and thus a high noise error rate, according to the testbed setup described in section VI-B. Recall that noise losses are measured via the loss rate for subsequent fragments. Fig. 10(a) plots the measured loss rate for first and second fragments on normal traffic and PIFS traffic. It can be seen that the loss rates are all similar, as expected in the absence of collisions and hidden nodes. This data also helps to confirm that the loss rate measured on second fragments is a good indicator of the noise loss rate experienced by other types of traffic.

As further validation of correct operation of the estimator, we classify the loss percentage of transmitted/received frames, respectively,

- $tx_{1,err} = (T_0 - A_0)/T_0$ i.e. the loss rate for first fragment transmissions



(a) Measured loss rate of first and second fragments and PIFS traffic.



(b) Cross-validation of measured noise loss rate.

Fig. 10. Measured loss rates for Low SNR link, no collisions, no hidden nodes. $tx_{1,err}$ is loss rate for first fragment transmissions, $tx_{2,err}$ loss rate for second fragments (an estimate of p_n), $rx_{1,err}$ the error rate measured at the receiver for first fragments, $rx_{2,err}$ the rate for second fragments.

- $tx_{2,err} = (T_S - A_S)/T_S$ i.e. the loss rate for second and subsequent fragments
- $rx_{1,err} = CRCerr_1/(R - I)$ i.e. the rate of CRC errors at the receiver for first fragments ($CRCerr_1$)
- $rx_{2,err} = CRCerr_S/(R - I)$ i.e. the rate of CRC errors at the receiver for subsequent fragments ($CRCerr_S$).

The measurement $tx_{2,err}$ is our proposed estimator for p_n , the frame loss rate due to noise errors. Note that the $rx_{1,err}$ and $rx_{2,err}$ measurements are obtained by an entirely independent estimator (operating at the receiver) from the $tx_{1,err}$ and $tx_{2,err}$ measurements (operating at the transmitter). As expected, Fig. 10(b) shows that the two estimators report very similar statistics for first and subsequent fragments, as the only errors present are noise errors⁴.

C. Hidden nodes only, no collisions, no noise

We now consider estimation of hidden node losses, again starting with a simple setup in this section in order to help gain clear insight into performance but with more complex situations considered in later sections.

Fig. 11 reports the experimental results for a setup with only one transmitter and one receiver (and so no collisions) and with one hidden node, the offered load at the transmitter and hidden

⁴For this validation, the receiver used fragment and retry bits to distinguish first and subsequent fragments. These bits may have been corrupted. Interestingly, despite uncertainty in these bits, the estimates are satisfactory.

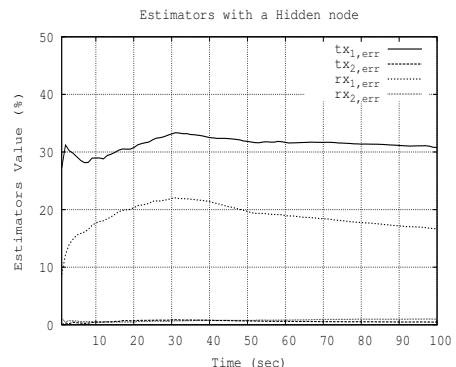


Fig. 11. Hidden nodes, clean channel, no collisions. $tx_{1,err}$ is loss rate for first fragment transmissions, $tx_{2,err}$ loss rate for second fragments (an estimate of p_n), $rx_{1,err}$ the error rate measured at the receiver for first fragments, $rx_{2,err}$ the rate for second fragments.

node being 300fps. As before, measurements at the transmitter are validated against independent measurements taken at the receiver. It can be seen that while the first fragment in a burst experiences a high error rate, the second fragment has a very low error rate. That is, as we expect, hidden node errors are limited to the first fragment sent in a burst, while second fragments are protected from these errors. It is interesting to observe that in this experiment the channel characteristics were slowly varying, as can be seen from the peak in loss rate after around 30s.

Note that the transmitter and receiver estimators report different error rates. This can be explained as follows: while measurements indicate that the number of CRC errors measured at the receiver is roughly the same as the number of retries measured at the transmitter, the number of busy slots is measured to be higher at the receiver because the hidden node's transmissions can be heard at the receiver. This test also confirms the different nature of the transmitter and receiver environments, and the need for a local estimator at the transmitter to take into account the point-to-point link quality characteristics.

D. Collisions and hidden nodes, no noise

Having validated the individual components of the estimator in basic scenarios, we now consider more complex situations with a mix of link impairments. In this section we consider a link with both collision losses and hidden node interference. In the experiments, the offered load at all stations is 300fps.

Firstly, we again use a setup with a pair of stations that behave as hidden nodes transmitting to one AP. Fig. 12(a) plots estimates of p_c , p_h , and p_n locally measured on one of the hidden node stations. It can be seen that p_h is estimated at a high value, as expected due to the severe hidden node interference in this example. The noise loss rate p_n is correctly estimated as being close to zero. The collision loss rate p_c is correctly estimated at a value very close to that measured with two contending stations and no noise or hidden nodes (marked as $p_c(p_h = 0, p_n = 0)$ in the figure, with the value taken from the measurements in Fig. 9). This is an encouraging result as it clearly demonstrates the ability of the proposed estimation

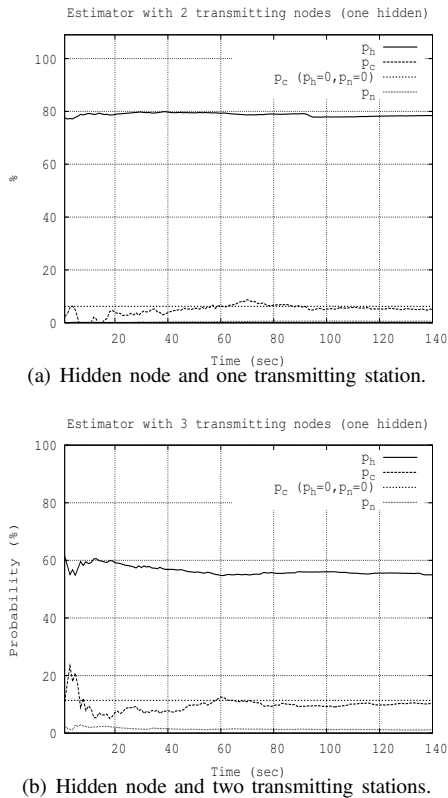


Fig. 12. Estimator values for p_c , p_h and p_n in the presence of collisions, hidden nodes and high SNR (low noise).

approach to effectively distinguish the different sources of frame loss, even under complex conditions.

Fig. 12(b) plots similar measurements, but now with a pair of stations that behave as hidden nodes plus one station which can be heard by all the other stations, for a total of three contending stations with saturated traffic. Again, the noise loss rate p_n is correctly estimated as being close to zero and the collision loss rate is correctly estimated as being close to that with three stations and no hidden nodes (marked on plot, with value taken from Fig. 9). The hidden node loss rate p_h is estimated at a high value, albeit somewhat lower than in the previous example (60% vs. 80%). This is caused by the third station transmissions, which are overheard by both hidden nodes, thus decreasing the number of hidden node transmissions and hence the hidden node probability.

E. Collisions, hidden nodes and noise

Finally, we consider a link suffering from all three loss inducing impairments: collisions, noise and hidden node interference. The scenario is illustrated in Fig. 13. We have three contending stations (stations 1, 2 and H), a pair of which behave as hidden nodes (stations 1 and H), and with a noisy channel between station 1 and its receiving station. Each station sends saturated traffic. Measurements gathered on station 1 are summarized in Fig. 14. It can be seen that the collision loss rate p_c is estimated at a value very close to that measured with three contending stations and no noise or hidden nodes (marked as “ $p_c(p_h = 0, p_n = 0)$ ” in the figure

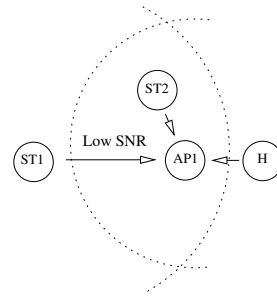


Fig. 13. Topology for hidden node with noise and contending stations.

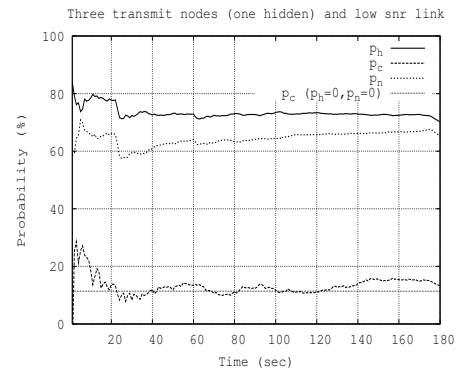


Fig. 14. Estimation with collisions, noise and hidden nodes.

with the value taken from the measurements in Fig. 9). That is, the estimator is able to successfully distinguish collision related losses from noise and hidden node related losses. It can also be seen from the figure that there is a high level of errors caused by noise and hidden node interference, with loss rates of approximately 65% and 75% respectively, providing a demanding test of our estimator.

F. Performance of RTS/CTS with hidden nodes

In this paper we use 802.11’s packet fragmentation functionality to mitigate hidden node effects. Of course, it is more common to consider the use of the RTS/CTS handshaking for this purpose and in principle the behavior should be similar. However, in practice we found a number of basic difficulties with the use of RTS/CTS for this purpose.

Firstly, consider an experiment with 7 stations transmitting traffic at 300fps without noise and hidden node interference. In Fig. 15(a) we plot the probability of collision with and without RTS/CTS (labeled as $rts - p_c$ and $no\ rts - p_{tot}$ respectively). The RTS/CTS collision probability is estimated from the number of missed CTS frames. To confirm the absence of noise interference, we have also plotted the overall probability of error (labeled $rts - p_{tot}$), which also takes into account the number of missed ACK over sent Data frame. Thus in this basic case, we see that RTS/CTS reliably estimates the probability of collision.

Now consider a scenario with a hidden node. As a baseline we collect data when two transmitting stations are within one another’s carrier sense region. As expected we see a low collision probability of around 7%, see Fig. 15(b) (line labeled $rst - no\ hi$). Now, we move the transmitters so that they

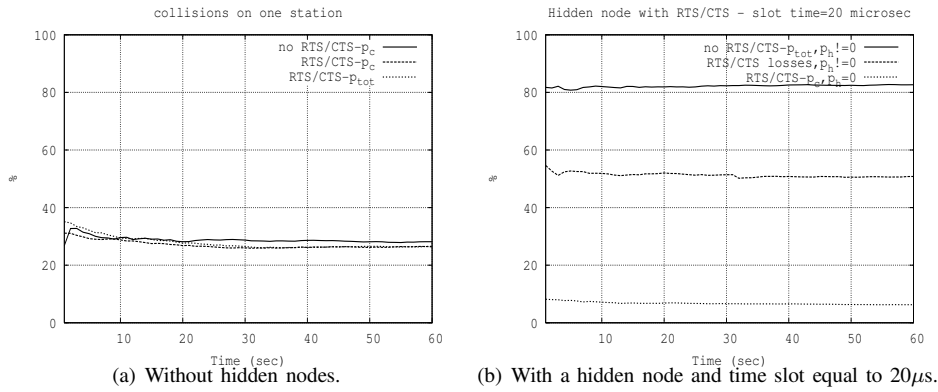


Fig. 15. Estimating p_c with RTS/CTS under various conditions.

are hidden from one another. In the absence of RTS/CTS, we measure a high error probability of around 82% (labeled *no rts*) which is mainly caused by hidden node errors. If we enable RTS/CTS, the error probability drops, but not to the expected value of 7%. Instead, we have a residual error of about 52% (line labeled *rts* in Fig. 15(b)). That is, in presence of hidden nodes the RTS/CTS estimator is still subject to considerable hidden node interference.

In order to understand this behavior, we note that the hidden node will defer its transmission if it overhears the CTS from the receiver before sending its frame. We can calculate when this occurs. Our tests used an 802.11g PHY. Station 1 sends an RTS frame (duration $48\mu s$), the receiver waits for a SIFS (duration $16\mu s$) and finally a CTS frame is sent by the receiver. Thus the hidden station would need to leave the medium idle for at least $64\mu s$ in order to receive the CTS frame. This is much longer than the PHY slot duration of $20\mu s$ for mixed mode 11b/g. Indeed if the backoff counter of the hidden node is less than 3 when the other station begins its RTS transmission, then the hidden node will make a transmission that corrupts the CTS frame.

VIII. ESTIMATING EXPOSED NODE AND CAPTURE EFFECTS

A. Exposed nodes

An exposed node is a sender station that senses the channel to be busy when, in fact, the channel at the receiver is idle and thus a successful transmission could have been made. A typical scenario is illustrated in Fig. 5. Here, *ST3* and *ST4* send data to *AP2* while *ST1* sends data to *AP1*. Sender *ST1* overhears the data transmissions by *ST3* and *ST4* and senses the channel to be busy. This is incorrect, however, since the physical separation between *ST3* and *ST4* and *AP1* means that transmissions by *ST1* would in fact be received correctly at *AP1* even when *ST3* and *ST4* are transmitting. *ST1* therefore defers its backoff countdown unnecessarily and its throughput suffers.

We implemented the topology in Fig. 5 in our testbed. *ST3* and *ST4* send 300fps traffic to Access Point *AP2*, while *ST1* uses the same channel to send at a low rate of 20fps traffic to *AP1* and station *ST2* 300fps to *AP1*. The channel is clean with no noise losses. In addition to measuring p_c , p_n and p_h

as before, we now also measure the total number of MAC slots R and the number I of slots which are detected idle. The value of $(R - I)/R$ is a measure of the proportion of slots which the MAC detects to be busy via carrier sense. The collision probability p_c provides a measure of the proportion of slots that are actually busy (in the sense that a transmission in that MAC slot would result in a collision). The difference between $(R - I)/R$ and p_c then provides a measure of how exposed a node is.

Our measurements for this situation are shown in Fig. 16(a). We show the collision probability p_c estimated using our technique and a fixed value measured without an exposed node (labeled " $p_c(1tx, p_{exp} = 0)$ "). It can be seen that these probabilities are low and close together. In this situation, measurements indicate that *ST1* senses the channel to be busy around 10% too often i.e. $p_{exp} = 10\%$. This suggests that *ST1* may freeze its backoff counter unnecessarily for 10% of MAC slots.

Fig. 16(b)–(d) show the corresponding measurements as the number of stations associated with *AP1* is increased. It can be seen that, as expected, p_c increases in line with measurements in Fig. 9 without exposed nodes. The exposed node probability p_{exp} is consistently measured as lying between 5% and 10%, although the relative impact of p_{exp} decreases as the number of stations increases.

To further explore our ability to sense exposed node effects, recall that exposed node effects are intimately related to the choice of carrier sense threshold used. In this scenario the carrier sense mechanism is too sensitive and *ST1* senses the channel busy too often. This effect is illustrated in Fig. 17 which plots the estimated p_{exp} vs. choice of carrier sense threshold for *ST1* in the setup of Fig. 5. As expected, it can be seen that the exposed node probability p_{exp} has the highest value for carrier sense thresholds in the range -90dBm to -80dBm . At around -75dBm , the value of p_{exp} decreases as the impact of *ST3* disappears (confirmed by inspection of packet traces). Finally, moving the carrier sense threshold up to -55dBm , the effect of *ST4* also disappears and *ST1* is no longer exposed (again, confirmed by detailed packet traces). Also shown in Fig. 17 is the measured collision probability p_c . It can be seen that this slightly increases as the carrier sense threshold is increased, which is to be expected as the

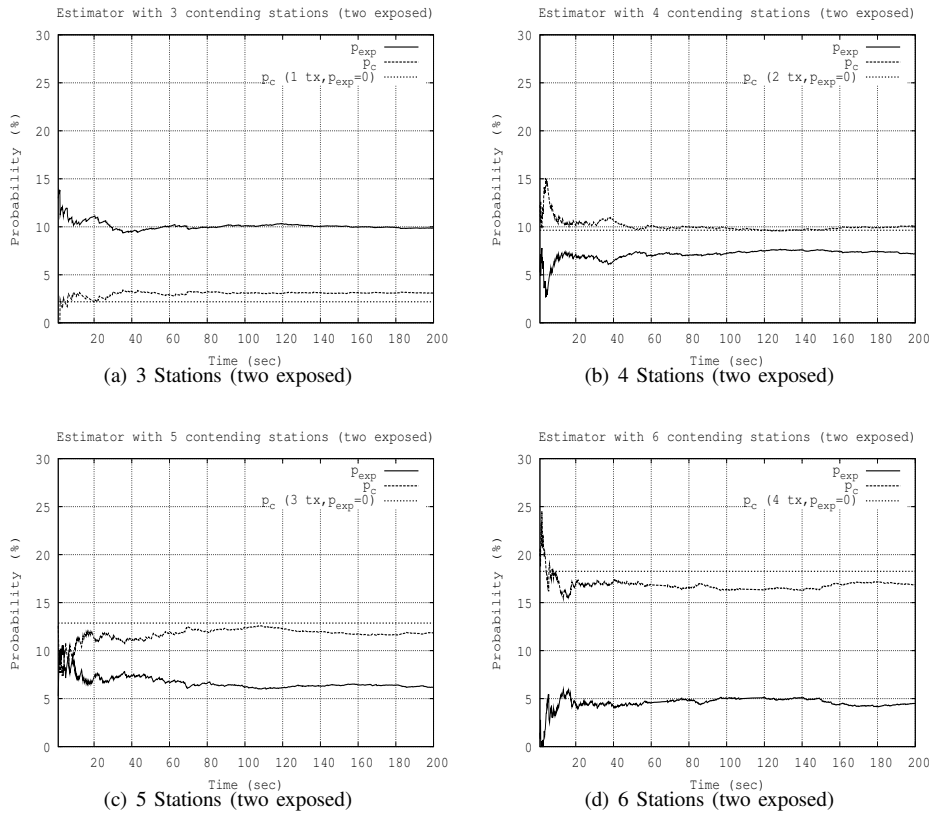


Fig. 16. Collision and exposed node probability vs. number of stations associated with *AP1*. Network topology as in Fig. 5.

backoff countdown of *ST1* is becoming of shorter duration. The benefits of using a suitable carrier sense threshold are illustrated in Fig. 18, which plots the *ST1*'s estimated MAC delay, which is the mean time between a packet arriving at the head of the interface queue and being successfully transmitted. The MAC delay is halved when the carrier sense threshold is increased to -55dBm instead of -85dBm .

A full carrier sense tuning algorithm would naturally be more complex and is beyond the scope of the present paper. However, this example does demonstrate the value and feasibility of being able to make this type of measurement.

B. Physical Layer Capture

Physical layer capture occurs when colliding transmissions have different received signal power. It may then happen that the transmission with highest power is successfully decoded even though it collides with another transmission. To assess the ability of our estimator to measure this effect, we configured our testbed as shown in Fig. 6. Station *ST1* sends data packets to *AP1* at a low rate of 20fps. In addition we have four other contending stations transmitting data to *AP1* at 300fps, but with lower received signal power than *ST1*.

Fig. 19(a) illustrates the impact of physical layer capture. It can be seen that *ST1* benefits from a lower than expected probability of collision. In particular, while with a total of five contending stations we expect a p_c of around 19% (based on measurements without capture) the measured collision rate at *ST1* is only around 8%. The difference of 11% is a direct

measure of the capture effect advantage experienced by *ST1*. To help validate the accuracy of this measurement, we took the same measurements with the carrier sense threshold increased to -60dBm — this change will not affect capture but would eventually highlight the presence of exposed node effects in our setup (see previous section). In Fig. 19(b), we find that the estimates of p_c and p_{plc} are almost unchanged, confirming the absence of exposed nodes in these tests.

We now further explore our ability to measure the impact of the capture effect. Note that decreasing the transmission power at *ST1* should reduce the capture effect. We confirm this experimentally in Fig. 20 which presents measurements of p_c and p_{plc} versus the transmit power at *ST1*. As expected, we see that the capture probability p_{plc} is greatest at the highest transmit power of 20dBm and that p_{plc} decreases to zero as the transmit power is reduced to 0dBm. Observe that, as might be expected, $p_c + p_{plc}$ remains roughly constant as the transmit power is varied, with a value around the expected probability of collision for five saturated stations.

Note that by reducing the transmit power a *ST1* we gain multiple benefits: power consumption is reduced, as is radio interference with adjacent WLANs, and the capture effect is removed, restoring fairness between contending stations. The effect on fairness of tuning the transmit power can be analyzed in more detail by looking at the probability of collision for each node in the network. We carried out tests with *ST1* transmitting at a low rate of 20fps plus four other stations with saturated traffic. Table II summarizes the experimental measurements obtained. We see that decreasing the transmit

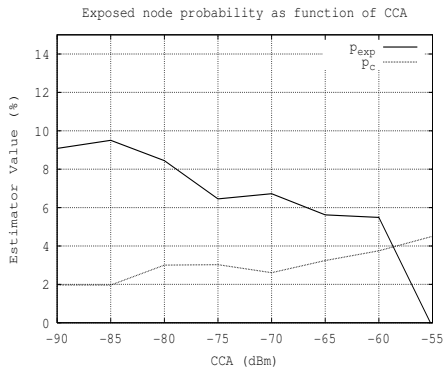


Fig. 17. Exposed node probability p_{exp} vs. carrier sense threshold.

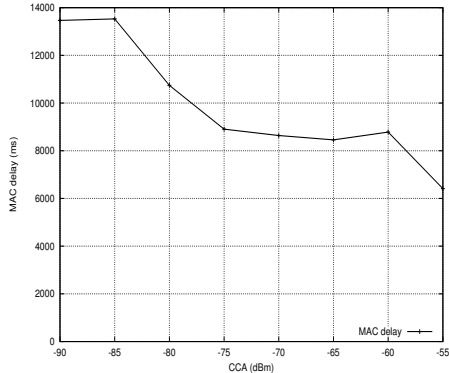


Fig. 18. MAC delay vs. carrier sense threshold.

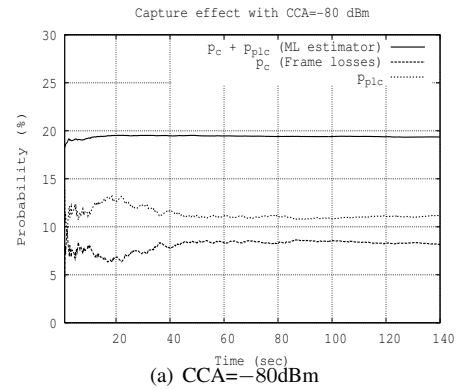
power at $ST1$ increases its the probability of collision. Meanwhile, the other nodes maintain a roughly constant collision probability p_c , thus improving fairness in the network. Note that p_c is not identical at all stations due to remaining capture effects at stations other than $ST1$ (power asymmetries arise due to antenna tolerances, differences in physical location, etc.). Adjustment of the transmit power at all stations could restore fairness.

	node 1		node 2	node 3	node 4	node 5
	TX power (dBm)	$p_c + p_{plc}$ (%)	p_c (%)	p_c (%)	p_c (%)	p_c (%)
16	18.8	2.3	14.9	11.0	17.3	15.9
13	18.4	5.5	13.6	12.4	18.1	16.3
10	18.0	9.9	14.5	10.9	17.6	16.1
7	17.6	11.9	14.3	12.3	17.3	16.0
4	17.5	15.6	12.1	12.7	17.7	16.1
1	17.5	17.1	14.1	10.6	17.8	16.3

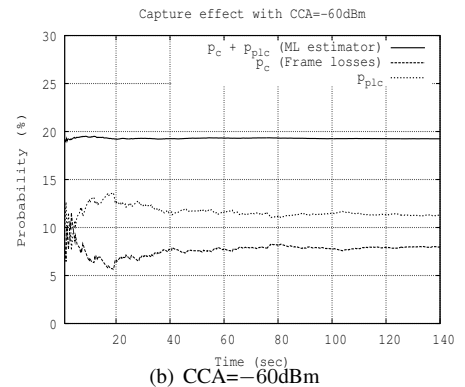
TABLE II
FAIRNESS WITH POWER TUNING.

IX. CONCLUSION

In this paper we propose a powerful MAC/PHY cross-layer approach to understanding transmission opportunities in 802.11 WLANs. Our approach provides separate quantitative measures of the severity of each type of impairment for the transmission, which may be incorporated into routing metrics or other higher layer statistics. Our estimator runs locally at the transmitter and it can be used to minimize the effect of transmission impairments by tuning transmission rate, carrier



(a) CCA=-80dBm



(b) CCA=-60dBm

Fig. 19. Demonstrating capture effect estimation. Results are shown for two different values of carrier sense threshold, to confirm the absence of exposed node effects in these tests. Network setup is as in Fig. 6.

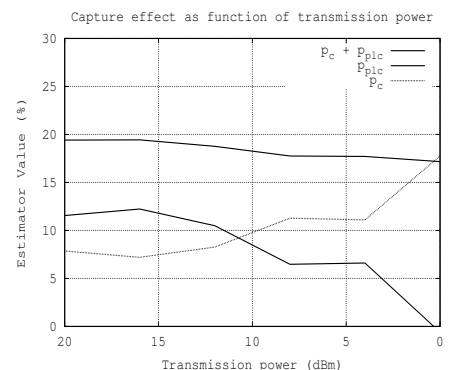


Fig. 20. Measurements of capture effect vs. transmit power.

sense, transmission power, and so on. We thus make available new measures that we expect to be of direct use for rate adaptation, channel allocation, etc. We explicitly demonstrate how the measurements can be applied in carrier sense tuning and power control and show possible benefits. Since we take advantage of the native characteristics of the 802.11 protocol — without requiring any modification to the standard or other stations — our approach is suited to implementation on commodity hardware and we demonstrate both a prototype implementation and experimental measurements. As the approach can be applied at a single transmitter to improve performance, there is a clear path to incremental deployment.

Our approach can be applied not only run-time at a node to optimize and select the rate, channel, carrier sense and trans-

mission power, but also offline to verify the presence/absence of impairments. This is a fundamental issue for network operators while troubleshooting and for researchers when establishing the realism and status of testbeds.

REFERENCES

- [1] C Reis, et al., “Measurement-Based Models of Delivery and Interference in Static Wireless Networks”, *SIGCOMM* 2006.
- [2] D Qiao and S Choi, “Goodput Enhancement of IEEE 802.11a Wireless LAN via Link Adaptation”, *Proc. IEEE ICC*, 2001.
- [3] D. Aguayo, et al. “Link-level measurements from an 802.11b mesh network”, *Proc. ACM SIGCOMM*, 2004.
- [4] I. Tinnirello, et al., “On the side-effects of proprietary solutions for fading and interference mitigation in IEEE 802.11b/g outdoor links”, *Elsevier Computer Network Journal*, February 2009.
- [5] I. Haratcherev, et al., “Hybrid Rate Control for IEEE 802.11”, *Proc. ACM, MobiWac*, 2004
- [6] K Ramachandran et al., “Scalability analysis of Rate Adaptation Techniques in Congested IEEE 802.11 Networks: An ORBIT Testbed Comparative Study”, *Proc. IEEE WoWMoM*, 2007.
- [7] S Wong, et al., “Robust Rate Adaptation for 802.11 Wireless Networks”, *Proc. ACM MobiCom*, 2006.
- [8] R Gummedi, et al., “Understanding and Mitigating the Impact of RF Interference on 802.11 Networks”, *SIGCOMM* 2007.
- [9] I Broustis, et al., “Implications of Power Control in Wireless Networks: A Quantitative Study”, *Proc. PAM*, 2007.
- [10] R Bruno, M Conti and E Gregori, “Mesh networks: commodity multihop ad hoc networks”, *IEEE Communications Magazine*, March 2005.
- [11] IEEE 802.11 WG, IEEE Std 802.11, 1999 edition “IEEE standard for Information technology — Telecommunications and information exchange between systems — Local and metropolitan area networks. Specific requirements. Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications”, 1999.
- [12] IEEE Std 802.11e, “Amendment to STANDARD [for] Information Technology — Telecommunications and Information Exchange Between Systems — LAN/MAN Specific Requirements — Part 11: Wireless Medium Access Control (MAC) and Physical Layer (PHY) specifications: Medium Access Control (MAC) Quality of Service (QoS) Enhancements”, 2005.
- [13] A Mishra, et al., “A Client-driven Approach for Channel Management in Wireless LANs”, *Proc. IEEE INFOCOM*, Barcelona, 2006.
- [14] B Kauffmann et al., “Measurement-Based Self Organization of Interfering 802.11 Wireless Access Networks”, *Proc. IEEE INFOCOM*, Anchorage, 2007.
- [15] G Bianchi, “Performance analysis of IEEE 802.11 distributed coordination function”, *IEEE J. Sel Area Comm.*, 18(3):535–547, Mar. 2000.
- [16] Rayanchu, et al., “Diagnosing Wireless Packet Losses in 802.11: Separating Collision from Weak Signal”, *INFOCOM* 2008.
- [17] A Kochut, et al., “Sniffing out the correct physical layer capture model in 802.11b”, *Proc. IEEE ICNP*, 2004.
- [18] D. Malone, et al., “MAC Layer Channel Quality Measurement in 802.11”, *IEEE Comms Let.*, 11(2):143–145, February 2007.
- [19] D. Giustiniano, et al., “Experimental Assessment of 802.11 MAC Layer Channel Estimators”, *IEEE Comms Let.*, 11(12):961–963, December 2007.
- [20] MAY Khan and D Veitch, “Isolating Physical PER for Smart Rate Selection in 802.11”, *Proc. IEEE 28th Conference on Computer Communications Infocom*, 1080–1088, April 2009.
- [21] A Sheth, et al., “MOJO: A Distributed Physical Layer Anomaly Detection System for 802.11 WLANs”, *MobiSys06*, June 2006.
- [22] J. Kim, et al., “CARA: Collision-Aware Rate Adaptation for IEEE 802.11 WLANs”, *Proc. IEEE INFOCOM*, 2006.
- [23] DJ Leith and P Clifford, “A Self-Managed Distributed Channel Selection Algorithm for WLANs”, *Proc. IEEE RAWNET*, Boston, 2006.
- [24] KJ Yu, et al., “A novel hidden station detection mechanism in IEEE 802.11 WLAN”, *IEEE Comms Let.*, 10(8):608–610, Aug. 2006.
- [25] P Acharya, et al., “Congestion-Aware Rate Adaptation in Wireless Networks: A Measurement-Driven Approach”, *IEEE SECON’08*, June 2008.
- [26] H. Ma, et al., “Joint transmit power and physical carrier sensing adaptation based on loss differentiation for high density IEEE 802.11 WLAN”, *Computer Networks*, June 2008
- [27] D. Malone, et al., “Modeling the 802.11 distributed coordination function in non-saturated heterogeneous condition”, *IEEE ACM T. Network*, 15(1):159–172, 2007.
- [28] J. Zhu, et al., “Adapting physical carrier sensing to maximize spatial reuse in 802.11 mesh networks”, *Wireless Communications & Mobile Computing*, Dec. 2004.

PLACE
PHOTO
HERE

including signal processing, channel quality and localization.

PLACE
PHOTO
HERE

Domenico Giustiniano Domenico Giustiniano received the M.Sc degree in Electronic Engineering from the University of Palermo in 2003 and the PhD in Telecommunication Engineering from the University of Roma Tor Vergata in March 2008. He spent 2007 as a visiting Ph.D. student at Hamilton Institute, Ireland. He was a Post Doc in the Internet Group of Telefonica Research, Barcelona from 2008 to 2010. He is currently Post Doc in the Wireless Communication Group of Disney Research, Zurich. His interests lie in the area of wireless systems,

David Malone David Malone received B.A.(mod), M.Sc. and Ph.D. degrees in mathematics from Trinity College Dublin. During his time as a postgraduate, he became a member of the FreeBSD development team. He is a research fellow at Hamilton Institute, NUI Maynooth. His interests include mathematics of networks, network measurement, IPv6 and systems administration. He is a co-author of O’Reilly’s “IPv6 Network Administration”.

PLACE
PHOTO
HERE

Doug Leith Doug Leith graduated from the University of Glasgow in 1986 and was awarded his Ph.D., also from the University of Glasgow, in 1989. In 2001, Prof. Leith moved to the National University of Ireland, Maynooth to assume the position of SFI Principal Investigator and to establish the Hamilton Institute (www.hamilton.ie) of which he is Director. His current research interests include the analysis and design of network congestion control and distributed resource allocation in wireless networks.

PLACE
PHOTO
HERE

Konstantina Papagiannaki Konstantina Papagiannaki received her first degree in electrical and computer engineering from the National Technical University of Athens (NTUA), Athens, Greece, in October 1998, and the Ph.D. degree from the Computer Science Department, University College London (UCL), London, U.K., in March 2003, receiving the Distinguished Dissertations Award 2003. She has been a Researcher with Intel Labs since January 2004: from 2004 until the end of 2006 in Cambridge, U.K., and since 2007 in Pittsburgh, PA. From the beginning of 2000 through 2003, she was a member of the IP Group at Sprint Advanced Technology Labs. She currently holds an adjunct faculty position with the Computer Science Department, Carnegie Mellon University, Pittsburgh, PA. Dr. Papagiannaki received the ACM SIGCOMM Rising Star Award in 2008.