Local Estimators for 802.11 MAC Channel Quality

Domenico Giustiniano, David Malone, Douglas J. Leith and Konstantina Papagiannaki

***Abstract*— We extend two approaches for estimating the proportions of frame losses at an 802.11 station due to collisions and other errors, in order to distinguish errors due to channel noise and hidden nodes. Our methods use local 802.11 measurements available in basic access mode and are based on MAC and PHY level measurements. We implement the estimators on experimental testbeds using off-the-shelf hardware to evaluate them in real wireless environments. We show that the estimators are effective and provide practical insight into the radio environment.**

I. Introduction

The CSMA/CA medium access mechanism of 802.11 makes estimation of channel quality challenging as frame loss due to collisions is a feature of normal operation. Importantly, the level of collision-induced loss is load dependent. The problem is how to disentangle collisions and losses due to channel impairment. Motivated by this, in [1] the authors propose a cross-layer approach implemented at the transmitter. A related technique is suggested in [2] for estimating the number of active stations. While these proposals were tested by simulation, they were not experimentally validated. The complex nature of the radio interference environment, and the limited accuracy of available channel models mean it is vital to evaluate the performance of these schemes on real hardware. A promising experimental demonstration of it was given in [3] and additionally introduced a second estimator based on counting numbers of CRC errors. While these estimators differentiated collisions and other errors, they made no attempt to say if other errors are due to noise or hidden node interference.

The contributions of this paper are threefold. First, we extend the the idle/busy estimator from [1] and the CRC estimator from [3] to consider a wider set of 802.11 network conditions. Second, we extended the idle/busy estimator to distinguish different channel impairments: channel noise and hidden nodes. Third, we present a refinement of the local CRC error technique from [3] at the 802.11 receiver for estimating the frame error rate due to radio interference that complements the idle/busy estimator. The extensions make use of 802.11's fragmentation feature. Our results use both estimators to allow cross-validation in a number of radio environments.

II. Related work

Most work on channel quality estimation has focused on PHY layer approaches based on SNR and RSSI measurements, e.g. in 802.11 see [4]. Due to a number of factors, including collisions, correlation between these and actual channel behaviour at the MAC layer may be weak (e.g. [5]).

We gratefully acknowledge the help of Richard Gass at Intel. Supported by Science Foundation Ireland grant IN3/03/I346.

In this paper we consider detecting the presence of hidden nodes. In [6], a station detects the existence of a hidden station if the medium has been idle over SIFS interval and an ACK or a frame of Length = 14 bytes is seen (i.e. either ACK or CTS) while CRC32 failure occurs from frame reception. In [7] the authors modify the 802.11 MAC to send NAK packets when a receiving station infers a channel error. This permits similar differentiation as our estimators at the expense of a non-standard MAC layer.

III. The Estimators

In this section we describe the two estimators. Both estimators identify hidden nodes via the 802.11 fragment scheme, which we briefly describe here.

802.11 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. 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 at the sender (by setting the 802.11 NAV field in the fragment) and at the receiver (using the NAV in the ACK).

The design of our estimators assumes that while a hidden node may not hear a transmitter, it is close enough to the receiver to decode the transmitted NAV value in ACK frames. Thus, in a burst of fragments, the first fragment will be subject to collisions, noise and hidden node errors, but subsequent fragments will only be subject to noise errors. Note that the 802.11e TXOP feature could be used instead of fragmentation, if the NAV field in the TXOP frames were appropriately set.

Our estimators also involve counting busy slots on the 802.11 medium. This makes the estimators dependent on carrier sense levels. In this paper, we assume that the levels are set so that no exposed nodes are present in the network. In an extension of this work we will deal with exposed nodes.

A. Idle/Busy Estimator at the Transmitter

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 slots. Consider operation from the viewpoint of a station, say station 1.

1) Station 1 has seen the medium as idle and, if backoff is in progress, has decremented its backoff counter. We call these *idle slots*.
2) 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.

3) Station 1 has transmitted and received an ACK. We call these slots *successful transmissions*.
4) Station 1 has transmitted, timed-out while waiting for an ACK and is about to resume its backoff. We call these slots *unsuccessful transmissions*.

As in [1], suppose that over some time period station 1 contends and transmits T_1 times and of these A_1 are successful because an ACK is received. We denote by p_c the probability of frame error due to collision and p_e the probability of frame error due to noise or hidden nodes. If station 1 transmits it will be successful with probability:

$$\mathbb{P}[\text{success}] = A_1/T_1 = (1-p_c)(1-p_e). \tag{1}$$

Suppose there are R slots in which station 1 does not transmit and that I of these are idle. The maximum-likelihood estimators for the collision and channel error probabilities are [1]:

$$p_c = \frac{R-I}{R}; \quad p_e = 1 - \frac{A_1/T_1}{1-p_c} \tag{2}$$

providing $1-p_c \leq A_1/T_1$. Note that these estimators are natural: collision probability p_c is estimated as the proportion of busy slots due to transmissions by other stations; p_e is then obtained by solving Eq. 1 for p_e once we know p_c.

We decompose p_e into hidden nodes and noise errors:

$$1-p_e = (1-p_h)(1-p_n), \tag{3}$$

where $p_h = \mathbb{P}[\text{error due to a hidden node}]$ and $p_n = \mathbb{P}[\text{error due channel noise}]$. In order to disentangle the contribution of p_n from p_h, we extend the estimator to collect statistics from 802.11 fragmented frames. As we noted, the first fragment of a burst can be lost due to collisions, hidden nodes and channel noise but subsequent fragments are protected from collisions and noise. So, while Eq. 1 is valid for the first fragment of a burst, for subsequent fragments:

$$\mathbb{P}[\text{success on subsequent fragments}] = A_S/T_S = (1-p_n), \tag{4}$$

where station 1 transmits T_S subsequent fragments and of these A_S are successful because an ACK is received. This gives an independent way to estimate p_n. Hence p_h can be estimated from Eq. 2, 3 and 4:

$$p_h = 1 - \frac{A_1/T_1}{(I/R)(A_S/T_S)}. \tag{5}$$

B. CRC-based Estimator at Receiver

We begin with the CRC-based estimator from [3]. The 802.11 frame consists of a PLCP (Physical Layer Convergence Procedure) preamble, a PLCP header and a Physical Service Data Unit (PSDU). Each PSDU consists of the MAC header, the frame body (MSDU) and of 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 header is correctly decoded but the MAC 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 have investigated the reporting of these errors on network cards based on the popular Intel 2915ABG and Atheros AR5213A chipsets. We find that CRC32 errors are reported accurately. On Intel cards we found that PLCP header errors were reported but preamble errors were not reliably logged. On Atheros cards, synchronisation errors could be reported multiple times for the same PLCP. Thus, we choose to work with the count of CRC32 errors for our estimator.

Consider when collisions, channel noise and hidden nodes result in CRC errors. First, note that in a collision two or more stations have chosen the same slot to start transmission. We assume that a third 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, a CRC error. We split p_c:

$$p_c = p_{c1} + p_{c2}, \tag{6}$$

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; p_{c2} collisions will be observed by the CRC estimator.

Second, consider channel noise. As the PLCP is usually sent at a substantially lower rate than the PSDU, we assume that channel noise always results in a CRC error (not PHY error).

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 a ordinary node select the same slot. These will contribute to p_c. However, hidden-node transmissions beginning in later slots (i.e., after a ordinary node has already started) may result in more complex errors. In our experiments we use 802.11g transmissions with a PLCP of 20μs and the 802.11b compatible slot length of 20μs. For this setup we expect all of the hidden node errors that are not simple collisions to result in CRC errors, as the hidden node will not transmit until after the PLCP has been transmitted.

Thus, the CRC errors seen at the receiver satisfy:

$$\frac{CRCerr}{R-I} = p_e + p_{c2} - p_e p_{c1} - p_e p_{c2} \approx p_e + p_{c2} \tag{7}$$

where $CRCerr$ is the number of CRC32 errors and $R-I$ is the number of busy slots seen at the receiver. Whenever fragmentation is applied at the transmitter, the receiver can maintain separate counters for first ($CRCerr_1$) and subsequent ($CRCerr_S$) fragments. This requires us to rely on PDSU fragment and retry bits, to determine if a fragment is part of a burst and if it is the first, or subsequent fragment of a burst. We read these bits, even though the PSDU may be corrupted. This lets us calculate the relative impact of p_n and p_h, though we cannot factor out p_{c2}. This is a weakness of the CRC-based estimator, but it is still useful for cross-validation and gaining insight into the radio environment.

IV. Implementation and Results

A. Implementation on commodity hardware and Testbed Setup

We have implemented these estimators using a combination of driver and firmware modifications to commodity network

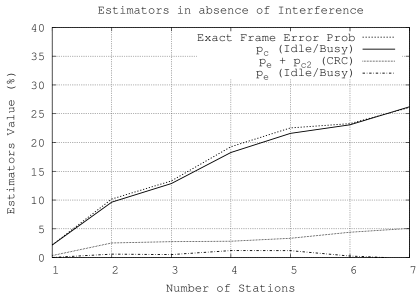

Fig. 1. Performance of the estimators for an interference-free station.

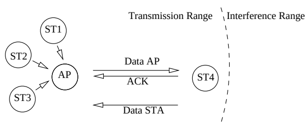

Fig. 2. Low SNR scenario.

cards using the Intel 2915ABG chipset. The measurement of transmissions T and success transmissions A (transmissions for which a MAC ACK is received) is straightforward at the driver level. However, the measurement of channel busy and idle times, needed to calculate R and I, requires carrier sense information from the hardware. We modified the card firmware and microcode to perform the necessary measurements and to expose these to the driver. For the CRC estimator, $CRCerr$ has been also retrieved from the microcode. We also use AR5213 Atheros cards for testing fragmentation issues.

We performed experimental measurements over a range of network conditions, of which we present a subset. Our testbed consists of Soekris net4801 devices running Linux and configured in infrastructure mode. Standard 802.11g parameters are used and antenna diversity is disabled in each node. Stations transmit 1400 byte UDP packets to an AP equipped with a NIC using the Intel 2915ABG chipset. The carrier sense threshold for the AP was set to -80 dBm. External interference is measured using a spectrum analyser.

B. Baseline without Hidden Nodes

1) Clean channel: Consider initially a situation with a clean channel and only a low level of external interference and channel noise (confirmed by spectrum analyser). All stations transmit packets to the AP at a rate of 300fps. Fig. 1 shows the estimated p_c and p_e as the number of stations is varied. For each configuration the estimation interval is 600s.

As a baseline, since the channel is clean, we can estimate the true frame error probability using $(T_1 - A_1)/T_1$. The collision probability p_c rises with the number of contending stations, as expected, and the p_c value from the idle/busy estimator is consistently close to the true frame error probability.

Also shown in Fig. 1 are the values of p_e and $p_e + p_{c2}$ as estimated by the idle/busy and CRC approaches respectively. We do not have accurate baseline measurements against which to evaluate the accuracy of these estimates, but note that the idle/busy approach uniformly estimates a low level of frame errors and that this is consistent with the clean channel

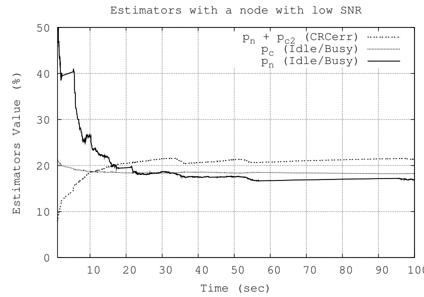

Fig. 3. Convergence of the estimator in presence of a node with low SNR.

conditions indicated by the spectrum analyser measurements. The CRC estimate of $p_e + p_{c2}$ is rather higher and increases with the number of stations. We believe that this is associated with the fact that CRC measurements are carried out at the receiver and thus affected by collisions that generate CRC errors, p_{c2}. An increased number of stations leads to an increased rate of collisions and so a greater p_{c2} value. While in qualitative terms we observe that the CRC estimator correctly identifies the channel as being good quality, we also see that the probability of collision leading to CRC errors, as represented by p_{c2}, may be an important factor.

2) Noisy channel: Next, consider a scenario where the channel quality is poorer, as in Fig. 2. All stations send traffic at 300fps to the AP. Stations 1–3 have good link quality. Station 4 is physically separated and has poor link quality (with SNR close to the receiver sensitivity for the rate of 12Mbps). The AP sends frames at a low rate (20 fps) to station 4. Fig. 3 shows time series of the estimated p_c and p_e for this poor-quality link. It can be seen that the converged estimate of p_c is close to that for 4 stations in Fig. 1, as expected. However, the idle/busy and CRC estimates of p_e and $p_e + p_{c2}$ are now much higher, indicating a frame error rate of around 20%. That is, both estimators are effective in distinguishing between collision and frame errors and correctly identify poor channel conditions. As in Fig. 1, the CRC-based estimate of $p_e + p_{c2}$ is higher than the idle/busy-based estimate of p_e. Again, we believe this is due to p_{c2} component of collisions.

C. Differentiating Channel Noise and Hidden Node Errors

Our hidden node scenarios are based on Fig. 6. We have a number of transmitting nodes and a receiver. The hidden node transmits to an independent receiver. We also verified that for our setup hidden nodes errors are seen as either simple collisions or CRC errors at the receiver.

1) Without Collisions: We now investigate the effectiveness of fragmentation for disentangling p_n from p_h. We look at two situations, the first with noise errors and the second with hidden node errors. First, consider a scenario with just one transmitter and one receiver, with a poor SNR, and fragmentation enabled at the transmitter. The transmitter sends 300fps to the receiver. The fragment threshold is selected so that two fragments of the same length are generated. We classify the loss percentage of transmitted/received frames, respectively $tx_{1,err} = (T_1 - A_1)/T_1$, $tx_{2,err} = (T_S - A_S)/T_S$ and $rx_{1,err} = CRCerr_1/(R - I)$ and $rx_{2,err} =$

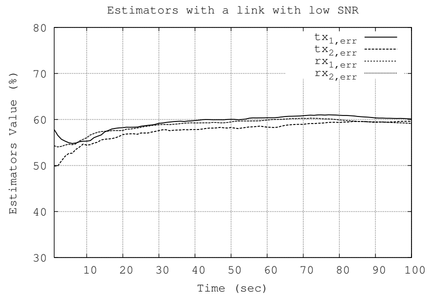

Fig. 4. Results with low SNR link and no hidden node.

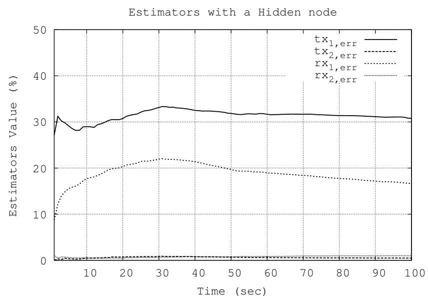

Fig. 5. Results with high SNR link and hidden node.

$CRCerr_S/(R-I)$. As expected for noise errors, Fig. 4(a) shows the two estimators report similar statistics for first and subsequent fragments. Note that the receiver used fragment and retry bits in the PSDU to distinguish first and subsequent fragments. These bits may have been corrupted, but our results indicate that the values are correct frequently enough that the estimators produce useful results.

Second, consider a similar setup, but with a hidden node, as in Fig. 6. Fig. 5(b) reports results for one transmitter, one receiver and one hidden node. The transmitter and hidden node transmit 300fps. While the first fragment in a burst experiences a high error rate, the second fragment has a very low error rate. As we expect, hidden node errors are limited to the first fragments, while the other fragments are robust to this issue. The transmitter and receiver estimators report a different fraction of errors. This can be explained as follows: while the number of CRC errors, is roughly the same as the number of retries at the transmitter, the number of busy slots is higher at the receiver because the hidden node's transmissions can be heard at the receiver. Finally note, in this experiment the channel characteristics were slowly varying, as can be seen from the peak in both estimates after around 30s.

2) With collisions: We analyse the impact of one hidden node plus 1–6 stations sending traffic to the AP for $p_n \approx 0$. We evaluate p_e on a selected link AP1–ST1, as a function of

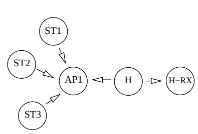

Fig. 6. Hidden node scenario.

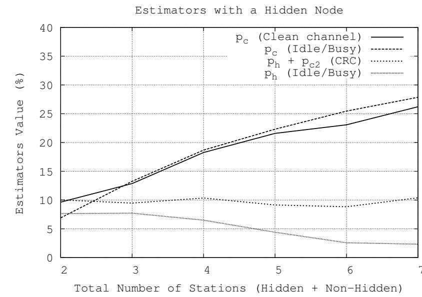

Fig. 7. Results for one hidden node.

the number of stations. Results are summarised in Fig. 7. It can be seen that the estimated probability of collision almost coincides with the value with no nodes were hidden (c.f. Fig. 1, also plotted on Fig. 7), as expected. However, p_h is now elevated for both the transmitter and receiver estimators. Note that p_h estimates at the transmitter decreases as the number of stations increases. On the other hand, p_h as estimated from CRC errors at the receiver remains almost constant as the number of stations is varied. This is due to the growth in p_{c2} with increasing number of stations. This contribution of p_{c2} to CRC errors is also responsible for the overall higher value measured at the receiver compared to the sender.

V. Conclusion

We implement and evaluate two local estimators for the probability of collision, frame errors due to noise and frame loss due to hidden nodes. We cross-validate these estimators, and find that they are effective in differentiating these types of frame loss. Our estimators make use of the different conditions experienced by first and subsequent frames in a set of fragments. In future work we aim to extend these methods further to also identify when the exposed node problem is present and apply these estimators to support decision making in algorithms for channel allocation [8], rate adaptation [9], [10], carrier-sense adaption, etc.

References

[1] D Malone, et al. "MAC Layer Channel Quality Measurement in 802.11", IEEE Comms Let., Feb. 2007.

[2] G Bianchi, I Tinnirello, "Kalman Filter Estimation of the Number of Competing Terminals in an IEEE 802.11 network", Proc. IEEE INFOCOM 2003.

[3] D Giustiniano, et al. "Experimental Assessment of 802.11 MAC Layer Channel Estimators", To Appear in IEEE Comms Let.

[4] D Qiao and S Choi, "Goodput Enhancement of IEEE 802.11a Wireless LAN via Link Adaptation", Proc. IEEE ICC, Finland, 2001.

[5] D Aguayo, et al. "Link-level measurements from an 802.11b mesh network", Proc. ACM SIGCOMM, Boston, 2004.

[6] KJ Yu, S Choi, K Jang, "A novel hidden station detection mechanism in IEEE 802.11 WLAN", IEEE Comms Let., Aug. 2006.

[7] Q Pang, SC Liew, VCM Leung, "Design of an Effective Loss-Distinguishable MAC Protocol for 802.11 WLAN", IEEE Comms Let., vol.9, no.9, Sep 2005.

[8] DJ Leith, P Clifford, "A Self-Managed Distributed Channel Selection Algorithm for WLANs", Proc. IEEE RAWNET, Boston, 2006.

[9] K Ramachandran et al., "Scalability analysis of Rate Adaptation Techniques in Congested IEEE 802.11 Networks: An ORBIT Testbed Comparative Study", WoWMoM 2007.

[10] J Kim, et al. "CARA: Collision-Aware Rate Adaptation for IEEE 802.11 WLANs", In Proc. IEEE INFOCOM 2006, Barcelona, Spain, April 23-29, 2006.