Estimating Contention of IEEE 802.11 Broadcasts Based on Inter-Frame Idle Slots

Quincy Tse*‡, Weisheng Si†, Javid Taheri*
*Centre for Distributed and High Performance Computing, The University of Sydney
†School of Computing, Engineering, and Mathematics, University of Western Sydney
‡Networked Systems, NICTA, Sydney, Australia
Email: {quincy.tse, javid.taheri}@sydney.edu.au
w.si@uws.edu.au

***Abstract*—Recent advances in communication technology has enabled vehicles to communicate with each other autonomously through the use of IEEE 802.11p protocol. Vehicle-to-vehicle communication regularly makes use of the broadcast mode transmissions, which are not often used prior to this application. Broadcast mode transmissions are more prone to frame loss from channel contention than unicasts due to its inability to adapt, and are unable to recover lost frames. This makes them very sensitive to channel congestion.**

In this paper, we first apply a variant of Bianchi *et al.*'s Markov model of the Distributed Coordination Function (DCF), to show that the observed inter-frame idle slots can be expressed as a probability distribution conditional on the number of saturated stations. It therefore follows that the probability distribution for the number of saturated stations can be estimated from inter-frame idle slots through Bayes Law. Second, we present a novel passive channel congestion estimation technique that observes the inter-frame idle slot counts in any given IEEE 802.11 network and uses a naïve Bayes classifier to estimate the current channel contention in terms of the number of concurrent saturated stations. This technique is evaluated using computer simulations, and is shown to produce more accurate estimates with faster convergence time than the existing technique of observing collision probability using channel busy status as a proxy.

I. Introduction

IEEE 802.11p had been proposed as the main underlying communication protocol for vehicle-to-vehicle communications. One of the main applications for this protocol is to provide a broadcast service for safety applications [1]. In the contention-based Medium Access Control (MAC) used in the IEEE 802.11 (including the amendment IEEE 802.11p), MAC-layer packet loss is highly dependent on the channel contention. This limitation is especially critical for broadcast-mode transmissions where explicit packet acknowledgements cannot be used, hence lost frames cannot be recovered automatically. Furthermore, the lack of explicit feedback also prevents the operation of the MAC-layer congestion control mechanism. Therefore, broadcast frames not only cannot be recovered automatically, they are also more prone to collisions.

Jiang *et al*. [1] identified the principle factor affecting CSMA/CA broadcasts as "communication density" ("number of carrier sensible events per unit area and unit time", *i.e.* product of message rate, station density, and range). For this reason, the ability to determine the current "communication density" (*a.k.a.* channel contention) can be very useful in improving packet reception. Once the current channel contention is estimated, the estimate may be used to adjust the contention window, allowing the MAC layer to adapt to channel congestion without the need of explicit acknowledgements. The estimates can also be passed up the protocol stack, allowing upper layer protocols (*e.g.* the retransmission scheme in [2]) to moderate the load it offers onto the channel in order to enhance the performance of the network.

This paper's contribution is twofold. First, we derive a technique to obtain an expression linking the number of idle slots between consecutive transmissions and the number of saturated stations based on a broadcast variant of Bianchi *et al.*'s Markov model [3]. Second, we exploit this relationship by using a naïve Bayes classifier, observing the inter-frame idle slots in order to estimate the channel congestion level in terms of the number of saturated stations on the network. Here, a station is "saturated" if it always has at least a frame in its transmit buffer. Note that only the legacy Non-QoS IEEE 802.11 Distributed Coordination Function (DCF) is considered in this paper. The presented technique can also predict the collision probability assuming ideal channels with no hidden terminals. Using computer simulations of the IEEE 802.11 DCF, naïve Bayes classifiers configured using this model is shown to be more accurate in estimating the channel contention, and converges to the steady state faster than the existing technique of observing channel busy status alone [4].

II. Related Works

The IEEE 802.11 DCF uses a variable-length contention window to in order to adapt to channel contention, thereby balancing channel utilisation and throughput. It assumes all packet losses are due to packet collision, and triggers exponential backoff when packet non-reception is detected. However, it can only work with unicast transmissions as broadcast-mode transmissions do not use acknowledgement frames.

To overcome the limitation of the IEEE 802.11 DCF contention sensing, Bononi *et al*. used "Slot Utilization" (SU), a metric that does not require close-loop feedback between stations [5]. SU is computed only when a station has something to send, and is defined as the fraction of busy slots observed

978-1-4799-0540-9/13/$31.00 ©2013 IEEE

during the station's initial backoff period. Unfortunately, SU had not been designed for continuous channel monitoring, and would result in potentially very slow updates.

A number of works have approached this problem theoretically, with the Markov model of the DCF presented by Bianchi *et al*. [3] being one of the most well known. Based on this model, Bianchi and Tinnirello [4] computed the collision probability as a function of the number of concurrent saturated stations. They have shown that the channel contention level, measured as the number of concurrent saturated stations, can be estimated by observing packet collisions, using the channel busy status as a proxy for packet collisions. (By assuming the observing station is saturated and is currently transmitting, the observation that the channel is busy means the observing station will cause a collision with the frames on the channel.) A range of different windowing and aggregation algorithms were developed based on this work, observing the channel busy status [4], [6]–[8].

Unfortunately the assumption that the observing station is going to transmit at every slot creates a "virtual station" that is over saturated — it transmits many more frames than a real saturated station. Furthermore, detecting packet collisions in wireless LAN is also non-trivial as packet losses could also be caused by physical layer effects as well. Unlike these techniques, the method presented in this paper observes the time between consecutive transmissions instead of packet collisions, and therefore does not rely on this assumption.

III. The IEEE 802.11 broadcast

The IEEE 802.11 basic access mechanism for broadcasts can be described as "listen-before-transmit". Here, we describe in detail the non-QoS-enabled basic access mechanism for broadcasts only. When a station has a frame to broadcast, it first waits for the channel to become idle for a period of longer than the Distributed Interframe Space (DIFS). This DIFS allows the higher priority control frames to be transmitted in preference to this new broadcast frame. Once the channel is sensed idle for a DIFS, the station additionally waits for a random backoff interval before transmitting. Without this backoff interval, stations with something to transmit will all transmit at the end of DIFS, causing frame collisions.

IEEE 802.11 divides the backoff intervals into periods of equal length called slots. When a station receives a frame from upper layers, an initial backoff counter is chosen from within the Contention Window (CW) $[0, aCW_{min}]$, where aCW_{min} is the initial CW size defined in the standard. During the backoff interval, if the channel is idle at the start of a slot and the backoff counter is zero, the frame is transmitted; if the counter is non-zero, the counter is decremented; otherwise, if the channel becomes busy, the counter is frozen until the channel becomes idle for DIFS again.

The above mechanism applies to both broadcast and unicast frames. Unicast frames additionally require explicit Acknowledgement frames to be transmitted by the recipient after frame reception in order to recovering lost (unacknowledged) frames. Stations may optionally employ RTS/CTS handshake to reduce the incidence of hidden terminals in unicasts. Broadcast frames cannot use Acknowledgement frames, thus lost frames are not retransmitted automatically. Furthermore, the basic access mechanism also describe the "exponential backoff" procedure for unicasts whereby transmitting stations would increase the CW in response to lost frames to reduce channel contention. The lack of Acknowledgement frames in broadcasts means CW cannot grow to adapt to the channel contention. Finally, broadcast frames cannot use the RTS/CTS procedure, making them more prone to the hidden terminal problem. For these reasons, broadcast transmissions are much more sensitive to channel contention, and the correct sensing of contention level is invaluable for moderating the offered load from upper layers, and for adapting the MAC behaviour dynamically.

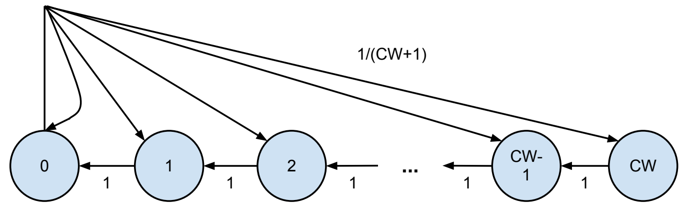


Fig. 1. Markov chain model for a single saturated station

IV. Markov model and steady state probabilities

The behaviour of the broadcasting non-QoS-enabled IEEE 802.11 DCF can be analysed by modelling its backoff counter using a variant of the Bianchi model [3] adapted for broadcast transmissions. This model uses only the top row of states in the Bianchi model, and discards the remaining states representing the exponential backoff procedure. Similar to the original model, this model quantises time into "slots" of varying lengths, delineated by the decrement of the backoff counter. Stations are assumed to be saturated (always have something to send) and are synchronised. In this model, we are not concerned with throughput or other time-related measures, therefore the length of each slot is unimportant. Each state in the model is represented by the value of the backoff counter at that state. Fig. 1 depicts this model graphically. If a backoff counter has a value of $X \in [1, aCW_{min}]$, it will always have a value of $X - 1$ at the next slot. If a backoff counter has a value of 0, the frame will be transmitted and the counter reset to a value uniformly distributed in the Contention Window $[0, aCW_{min}]$ as per IEEE 802.11 specifications.

Solving the system of probability equations, the steady state probabilities for each state can be derived. This resultant formula (Eqn 1) represents the overall probability that a station's backoff counter has a value of k.

$$P_k = \frac{2(CW + 1 - k)}{(CW + 1)(CW + 2)} \tag{1}$$

Unlike in Bianchi and Tinnirello's approach [4] where they solved the expression for collision probability, we attempt to determine the number of backoff slots between transmissions. Based on the derived probability distribution for a station's backoff state, the probability distribution of idle slots between consecutive transmissions can be derived. However, since channel observations are not taken in a process that is independent from the underlying states (observations only occurs after reaching state 0), a simple binomial expansion of P_k does not yield a correct result. This is especially evident when the number of stations (N) is either to small or too large compared to the CW size. The ideal model would account for the entire history of the station, but would therefore become intractable.

In order to reduce the error due to sampling whilst allowing the model to be solved in reasonable time, one-step history can be added to the model. In this model, we denote the uniform distribution over the contention window $[0, CW] \subset \mathbb{Z}$ as U, and the probability of choosing slot k from this distribution $U_k = \frac{1}{CW+1}$. The conditional probability of having n concurrent transmissions at slot k given there was no transmissions in any of the previous slots can therefore be written as Eqn 2.

$$ {}^{n}T_{k|{}^{0}T_{k-1}} = \sum_{i=1}^{N} {}^{i}T \sum_{m=\max(0,n-N+i)}^{\min(i,n)} \binom{i}{m} (U_k^*)^m (1-U_k^*)^{i-m} \binom{N-i}{n-m} (P_k^*)^{n-m} (1-P_k^*)^{N-i-n+m} \tag{2} $$

where:

$$ U_k^* = \frac{U_k}{1-\sum_{j=0}^{k-1} U_j} $$

$$ P_k^* = \frac{P_k}{1-\sum_{j=0}^{k-1} P_j} $$

This model accounts for the fact that stations who transmitted in the previous time slot will have its backoff counter reset to a value uniformly distributed across the CW. It is assumed that the remainder of the stations have reached steady state, hence follow the solution of the Markov model. In Eqn 2, i represents the number of concurrent transmissions in the last transmission cycle, and ${}^{i}T$ is the probability of this happening in the steady state. m is the counter variable representing the number of stations transmitting at the current slot and also transmitted in the previous time slot. This expression is simply a summation of binomial expansions over all permutations of possible number of stations previously transmitted.

From this expression, the unconditional probabilities ${}^{n}T_k$ can be determined by Eqn 3, the probability of n concurrent transmissions by Eqn 4, and the idle slot probability distribution by Eqn 5.

$$ {}^{n}T_k = {}^{n}T_{k|{}^{0}T_{k-1}} \prod_{j=0}^{k-1} {}^{0}T_{j|{}^{0}T_{j-1}} \quad \text{where } {}^{0}T_{-1} = 1 \tag{3} $$

$$ {}^{n}T = \sum_{k=0}^{CW} {}^{n}T_k \tag{4} $$

Algorithm 1 Estimating the number of saturated stations

Let $\boldsymbol{b}$ be the belief vector of saturated stations count.
for all $b_i \in \boldsymbol{b}$ **do**
  $b_i \leftarrow \frac{1}{\mathrm{length}(\boldsymbol{b})}$
end for

loop
  $\hat{o} \leftarrow$ observed number of idle slots
  $denominator \leftarrow 0$
  for all $b_i \in \boldsymbol{b}$ **do**
    $denominator \leftarrow denominator + b_i(T_{\hat{o}}|N = i)$
  end for
  for all $b_i \in \boldsymbol{b}$ **do**
    $b_i' \leftarrow \frac{b_i(T_{\hat{o}}|N=i)}{denominator}$
  end for
  $\boldsymbol{b} \leftarrow \{b_i' \quad \forall i\}$
end loop

$N_{est} \leftarrow \sum_i ib_i$ ▷ Estimate is the weighted sum of belief

$$ T_k = 1 - {}^{0}T_k \quad k \in [0, CW) \in \mathbb{Z} \tag{5} $$

Determining the solution to the model requires solving a system of polynomial equations of N variables and degree CW, and is difficult to solve analytically. For this reason, we calculated the solutions numerically. Solutions for up to 450 stations and contention window size of up to 255 were computed.

V. Channel Contention Estimation

Since the probability distribution of idle slots between transmissions can be determined as a function of the number of concurrent saturated stations, one can therefore estimate the number of saturated stations based on the observed distribution of idle slots.

A naïve Bayes classifier [9] can be used to estimate the number of saturated stations based on the channel observations. Naïve Bayes classifiers are based on Bayes theorem in probability theory such that observed outcomes are used to derive a distribution of the underlying factors on which the observations are conditional. Such classifiers are simple to implement, and are known to be quite accurate in practice. Algorithm 1 outlines the operation of a naïve Bayes classifier.

The resultant belief vector $\boldsymbol{b}$ represents the likelihood that the current estimate is the correct number of saturated stations in the network. One method to interpret this belief vector is by taking the entry with the highest likelihood (Maximum Likelihood). However, since the number of categories used in this classifier is much smaller than the domain of the conditions, we took the weighted sum of the belief vector as the estimated contention value. This allows the classifier to interpolate for the number of saturated stations that is not in the referenced set.

VI. Performance Evaluation

A. Accuracy of DCF model

Computer simulations were conducted using a simplified model of the DCF to verify the correctness of the model. Complex simulation packages like ns-3 were not used in the investigation because these packages also simulate more complex physical layer effects that complicates the interpretation of results. Simulations were conducted for CW sizes of 3, 7, 15, 31, 63, 127 and 255, with up to 100 saturated stations within range of each other. From the simulated executions of the stations' DCF backoff counter states, statistics on collision probability and idle slot count between transmissions were collected. These statistics were then compared to the model predictions.

1) The simulation model: A model simulating a DCF backoff counter was constructed to verify the mathematical model. This simulation model is a simple decrementing counter that is reset once it reaches zero, and assumes perfect physical channel. The model simulates the following:

- Fixed size contention window (CW) for each station;
- Backoff counter reinitialise to a uniformly distributed value within the CW after transmission by the station. This models the DCF broadcast behaviour (*i.e.* no ACKs) and assumes all stations are saturated;
- Global (shared) timeline in "slots". Data transmission, IFS, *etc.* occur between slots and the actual wall time for the action is ignored;
- Transmission is lost if and only if there is a collision (two or more stations scheduled to transmit in the same slot); and
- Model assumes all stations are synchronised (propagation and processing times are zero and no hidden stations). Without assuming synchronisation, we cannot ignore the time between slots as stations that are not synchronised will see different slot boundaries.

Each station simulated is initially assigned a backoff counter value uniformly distributed over the contention window. At each time step, all backoff counters are decremented by one if the counter value is greater than zero. If the counter is zero, it is assumed that the station will initiate a transmission, and the counter is reset to a backoff counter uniformly distributed over the contention window. The transmission is assumed to be successful if only one station initiated a transmission, and assumed to have failed due to collision if more than one station transmitted. If no station initiated a transmission at that timeslot, then the channel is considered idle at that time, otherwise, the channel is considered busy. In this simulation, statistics on idle periods, probability of channel being busy and packet success ratio are collected.

This model uses the Combined Multiple-Recursive Generator MRG32k3a as the pseudo-random number generator. This generator is also used in many complex network simulation packages like ns-2 and ns-3. Each execution of the model uses a different seed, thus has a good probability that the executions are statistically independent from each other. In the experiments, a sample of 500,000 idle periods was collected for each station count–CW pair.

TABLE I
Overall network statistics goodness-of-fit

CW	15		63		255	
	Slots	Error	Slots	Error	Slots	Error
R^2	0.9989	0.9976	0.9999	0.9999	1.0000	1.0000

B. Estimating channel load

Networks containing a fixed number of saturated stations, each operating as described were simulated. During these simulations, channel observations are fed to two naïve Bayes classifiers, with the estimates from the classifiers compared against the parameters of the simulation. The experiment compares channel estimation by observing inter-frame spaces against Bianchi and Tinnirello's approach [4] of observing channel busy status at each time slot in terms of both accuracy and the time to reach steady state.

In this experiment, one of the Bayes classifiers is configured with the probability distribution of idle slots calculated by the theoretical model, and are given the idle slot observations every time a transmission occurs. The other Bayes classifier is configured with the steady state probability of transmission in any given time slot. The estimator configured to observe channel busy status is fed channel observation (busy or not) at every time slot.

It is noted that, unlike in Bianchi and Tinnirello's paper [4], a naïve Bayes classifier was used. Not only is a naïve Bayes classifier very easy to implement, it also provides a common basis to compare the two approaches. The categories for the classifiers were chosen to be the data points originally obtained for the previous section (Fig. 3), (including data points collected but not plotted in the figure). The data points were chosen mainly due to convenience and are not the categories that optimise for classification accuracy, or usefulness in prediction.

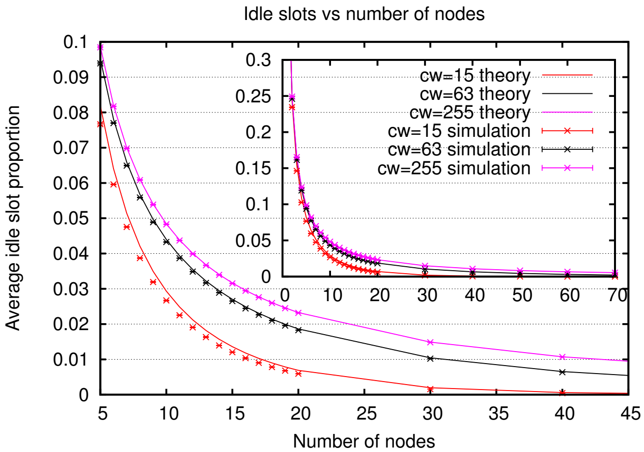
VII. Results

A. Model accuracy

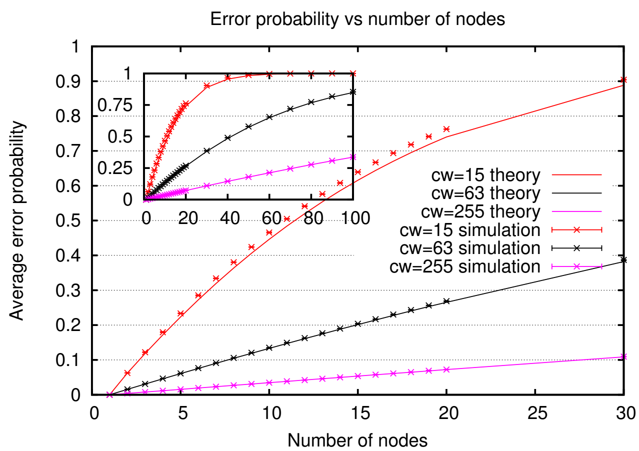
Fig. 2 compares the overall network statistics between the model prediction and the simulation results. Further comparison looking at the distribution of backoff slot for a contention window of 63 and varying number of stations are included in Fig. 3.

The simulation results (Fig. 2) suggest that the expected number of idle slots decreases and the collision probability increases as the number of saturated stations increases. This confirms the intuition that as more stations tries to transmit, the chance that some stations would transmit while another is still decrementing the backoff counter increases. This also confirms the intuition that when the number of stations increases, the chance of two of more stations choosing the same backoff slot also increases.

By comparing the overall statistics from the theoretical prediction to the simulation output in Fig 2, the accuracy



(a) Mean idle slots proportion

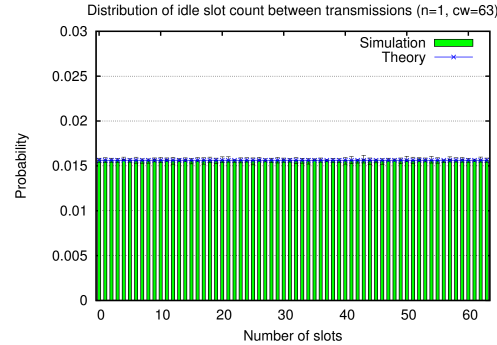


(b) Mean proportion of packet failures due to collisions

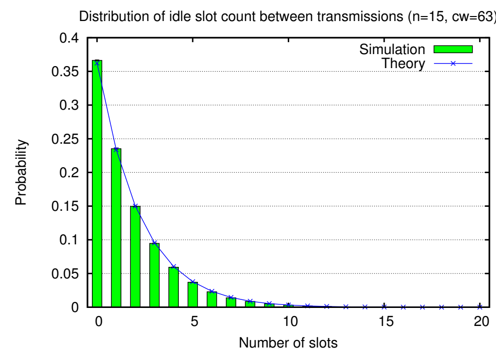
Fig. 2. Overall network statistics for a contention window size of 64 as a function of total number of concurrent saturated stations, in the absence of hidden stations, as predicted by the theoretical model vs simulation results. Simulation results are aggregated over 10 executions of the simulation using different random seeds. Error bars denotes the range of values observed.

of the theoretical model in predicting the expected idle slot counts and the associated error probabilities can be confirmed graphically. In addition, Table I calculates the R^2 value for the model. Both the idle slots and the success/error predictions have R^2 close to 1, indicating very high correlation between the observed data and the model.

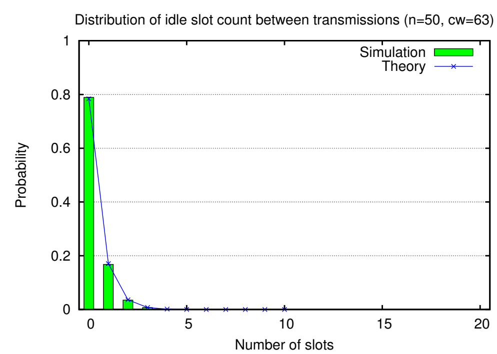
Fig. 3 further compares the performance of the model with the simulation, looking at the probability distributions when the contention window size is restricted to 64. In general, these plots indicate that the theoretical model presented is quite accurate in predicting the probability distribution of idle slot counts. Fig. 3c showed that the theoretical model very slightly underestimates the probability of immediate transmissions at very high station densities (50 stations in range). This small discrepancy would explain the underestimation of packet loss observable in Fig. 2b. Table II shows the R^2 values for these predictions. All results except for $N = 1$ shows a very high R^2 value, indicating high correlation between the prediction



(a) 1 station (y-axis range from 0 to 0.03 only). ($\frac{1}{64}$ = 0.015625)



(b) 15 stations



(c) 50 stations

Fig. 3. Distribution of idle slots between transmissions (inter-frame space) for a contention window size of 64, as predicted by the model (line) vs simulation results (columns) using the simplified DCF model. Simulation results are aggregated over 10 executions of the simulation using different random seeds. Error bars shows the range of values.

TABLE II
GOODNESS-OF-FIT OF PREDICTED DISTRIBUTION ($aCW_{min} = 64$)

#Nodes	1	5	10	15	50	150
R^2	1.4e-11	0.9998	0.9999	0.9999	0.9999	1.0000

and the observations. For the case $N = 1$, since the prediction is a horizontal line, the R^2 value cannot to provide a useful measure of correlation. Nevertheless, the good fit between the prediction model and the data can be confirmed visually using graphical means.

B. Load estimation results

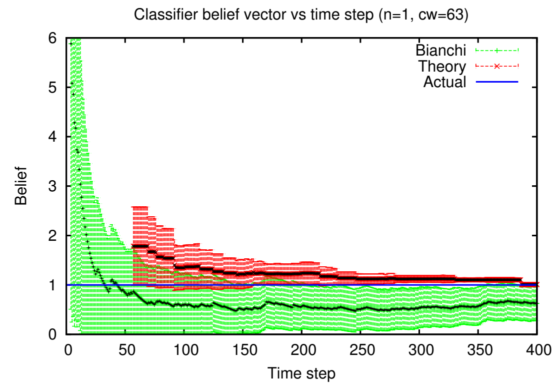
Fig. 4 plots the respective belief vectors from the two naïve Bayes classifiers. It should also be noted that Bianchi and Tinnirello's approach assumes the observing station is always transmitting (saturated), thus the actual output from this classifier is one higher than the number of saturated stations. Fig. 4 and Fig. 5 have been adjusted to account for this behaviour.

Overall, the naïve Bayes classifier that observes idle slot counts outperforms the one observing the channel busy status ("collision probability"), in terms of both estimation accuracy and time to steady state. Fig. 4 shows the mean and standard deviation of the probability distribution in the belief vector over time for a system with contention window size of 64 ($aCW_{min} = 63$). The number of saturated stations tested include the case with only a single station (Fig. 4a), less stations than the number of slots available (Fig. 4b), and the number of stations close to the number of slots available (Fig. 4c). The case when the number of stations exceeds the number of backoff slots available was also tested showing similar output to the one with 50 stations, and is not presented in this paper for sake of brevity.

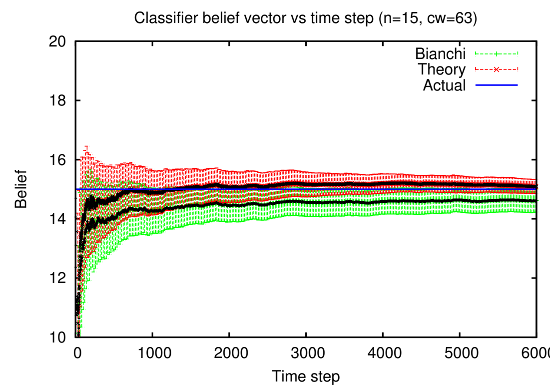
Fig. 4 show that both Bianchi and Tinnirello's [4] and our techniques can be used in a naïve Bayes classifier to determine the number of saturated stations based on channel observations. The black points marks the mean of the belief vector, and the error bars shows the spread of the probabilities (one standard deviation). The red error bars are the belief vectors from the classifier observing idle slot counts, whereas the green error bars corresponds to those observing channel busy status. These figures showed that as time progresses, the means of the belief vectors for both techniques converges to a value close to the actual parameter (*i.e.* classifier estimates are accurate), and the spread of the probabilities reduces (*i.e.* the classifier is becoming more certain).

It is noted that the early data points for the classifier observing idle slot counts is outside the range of the y-axis in Fig. 4a. This is due to the fact that, unlike observing channel status, the belief vector cannot be updated until the first transmission when observing idle slot counts, and it may take a few observations before the belief vector gets within the range of the y-axis.

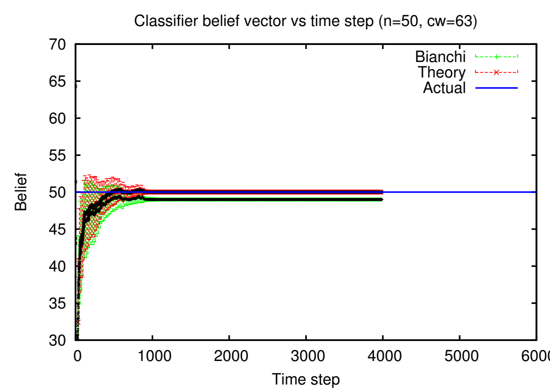
In the scenarios tested, the classifier that observes idle slots tends to converge to a value closer to the true simulation parameter. The mean estimate from Bianchi and Tinnirello's technique tends to be half to one station lower than the actual parameter. In addition, the classifier observing idle slots also are slower to become certain about the estimate, as can be seen by the higher spread.



(a) 1 saturated station (x-axis range up to 400 steps only).



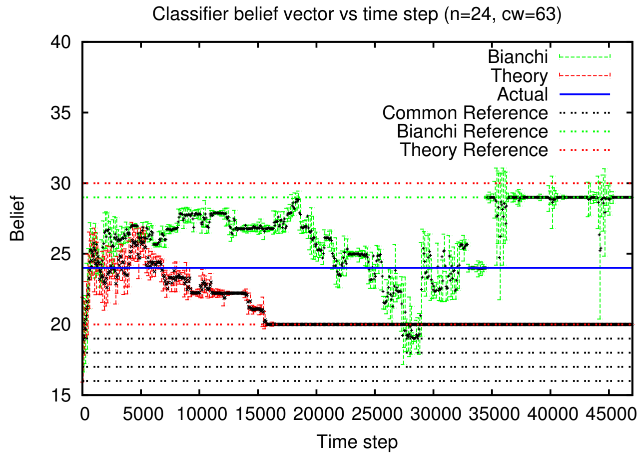
(b) 15 stations



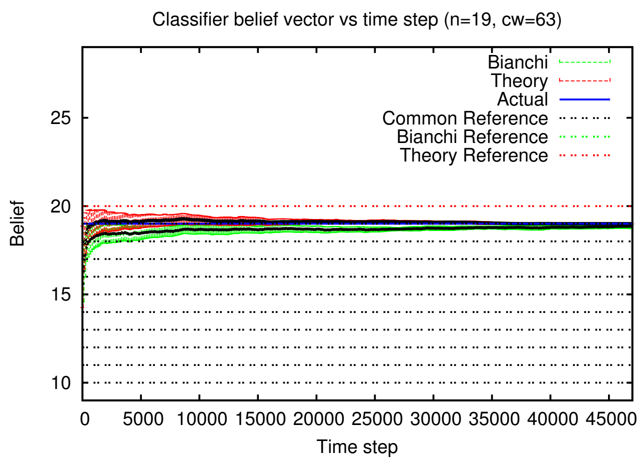
(c) 50 stations

Fig. 4. Classifier belief vector as time progresses for a contention window size of 64. Plotted is the mean and standard variation of the belief vector probability distribution. The blue solid line represents the true configuration of the simulation. Only one in 20 data points are plotted to allow the other line to show through. Early values for the red "theory" points (from the idle slot observation method) are outside the plot until the after the first backoff period — *i.e.* the first belief vector update.

1) Estimating non-referenced values: When the number of saturated stations are not in the set of reference values, the classifiers is likely to eventually choose as result a member of the reference set instead of the true value. Fig. 5 compares the belief vectors between (a) a non-referenced number of saturated stations (24, the closest categories are 20 and 30),



(a) 24 saturated stations — not in reference set



(b) 19 stations — an element of the reference set

Fig. 5. Comparison of estimated contention levels when the actual number of saturated stations is either (a) outside the reference set, or (b) an element of the reference set. Estimated contention level is plotted as the weighted sum and the standard deviation of the classifier belief vectors. Contention window size of 64 was used. Only one in 70 points are plotted to allow other error bars to be visible. Plot shows a much noisier output when contention level is not within the reference set, with the output converging to a value in the reference set instead of the true value.

and (b) with 19 saturated stations, which is an element of the reference set. In the figures, the reference values common to both classifiers are coloured black, ones unique to Bianchi and Tinnirello's technique (due to the off-by-one behaviour) are green, and the ones unique to the idle slots approach are coloured red.

Fig. 5 shows that when the number of saturated stations is not within the reference set, the naïve Bayes classifier output may eventually converge to a value within the reference set instead of the true value, with the output from the classifier being much noisier. Consistent with earlier observations, the estimates for 19-station test (Fig. 5b) converges consistently to the correct value for both estimators. In comparison, the 24-station test (Fig. 5a) converges much slower to the steady state, with the mean lingering around the true value for an extended period of long time before stabilising at a referenced value. The classifier observing channel busy status showed occasional large variance in its estimates even after the mean value reached its steady state.

VIII. Discussion and Future Work

The investigation in this paper raises some concerns regarding the capacity of IEEE 802.11p for vehicular safety systems. The expected capacity of the current IEEE 802.11p configuration (where aCW_{min} is 15) is obvious from the $CW = 15$ plot in Fig. 2b. Current proposals for vehicular safety applications require a minimum packet reception ratio of 90%. However, even two saturated stations on the network will already degrade the PRR to the threshold without considering physical effects that contributes to frame loss! The investigations presented in this paper suggest that, in order for vehicle-to-vehicle communication to meet the target PRR, offered load to the channel must be tightly controlled. Enabling the Medium Access Control function to sense and adapt to channel load using non-explicit feedback mechanisms like passive channel observations may be necessary.

In terms of channel observation techniques, it is observed that the output from Bianchi and Tinnirello's technique is noisier. The noise is more prominent with smaller contention window sizes probably because the technique only uses the binary busy/idle status to estimate channel condition. This means individual states may be much more influential on the mean. It also explains why idle slots observations generate a less noisy result. The wider range of possible outcomes from observing idle slots is also beneficial for improving confidence in the estimates, thus allows the estimates to converge faster despite the slower update frequency.

This work also highlights the need for appropriate windowing strategies or the use of more sophisticated classification/regression algorithms. Naïve Bayes classifiers retain infinite history, therefore it cannot track changing channel conditions. Retaining infinite history means that when the number of samples is large enough, additional samples would provide minimal influence on the classifier unless an extremely rare event is observed. Fig. 5a showed that as time progresses, the classifier output converges to one of the reference values due to the lack of windowing strategy. Experimental results suggested that the estimator output may linger near the true value for some time before converging to the steady state value. Hence an appropriately chosen aggregation window (size and/or shape) could potentially avoid the classifier getting stuck at a certain category, and allows the classifier to track the changing condition.

The use of naïve Bayes classifiers in this paper was only intended to be a sample application of the model. The theoretical model presented a method to compute the expected number of idle slots that can be expected for a certain number of saturated stations in the network. This work also demonstrated that one may use observed idle slot counts to estimate the channel load in terms of the number of saturated stations. In actual practice, one can use the presented model and the technique of observing idle slots in estimators other than (or in conjunction with) a naïve Bayes classifier to improve the accuracy and/or time to steady state. One may, for example, apply a hamming window, ARMA and/or EKF filter [4], MAP filter [6], or Viterbi algorithm over the output of the Bayesian estimator [8]; or to use particle filter techniques in place of the Bayesian estimator similar to [7]. The work presented in this paper can be used as the basis of any applicable classification/regression techniques in order to estimate channel load.

Furthermore, when compared to Bianchi and Tinnirello's technique [4] of observing frame collisions, observing idle slot counts converges faster despite the slower refresh rate. Observing channel busy status causes the estimator to slowly adjust its belief vector at every slot, whereas observing idle slots cause large adjustments every few slots. Since channel contention is unlikely to change much between slots, the slower refresh rate does not affect the usefulness of the technique. On the other hand, the faster convergence enables the use of smaller aggregation windows, thereby allowing classifiers that does not retain full history to track current channel contention quicker in a dynamic environment.

Finally, in modelling this system, all stations are assumed to be within range of other and saturated. This is atypical in real life. When a station cannot sense a current transmission, the assumption that all stations are synchronised becomes invalid, thus may invalidate the model. Intuitively, hidden terminals might transmit during another station's transmission, thus one may no longer disregard the timing aspect of the scenario, and cannot use flexible slots as the unit of time (without assuming transmission takes integer number of slots). Additionally the state transition of one station is no longer independent of another station. For these reasons, a model that allows hidden terminals cannot be Markovian in the current form. Bastani *et al*. used a slight variant of this Markov model whereby time is still quantised into slots, but the DCF state may not update at each slot depending on the channel condition [10]. A similar extension of the model presented may be useful for incorporating hidden terminals. Further investigations on the actual effect of both non-saturated stations and hidden terminals on idle slot distribution is needed.

IX. Conclusion

In this paper, we have applied the Bianchi model to derive an expression relating the number of idle slots between IEEE 802.11broadcast transmissions, to the number of saturated stations on the network. We then presented a channel contention measurement technique exploiting this relationship. The presented technique uses naïve Bayes classifiers to observe idle slot counts between frames. Through computer simulations, we have shown this technique to be effective in estimating the number of saturated stations on a network with no hidden terminals. When compared to the existing technique of observing packet collision probability, the technique of observing idle slot counts reaches steady state faster, with the estimate being closer to the true value.

Acknowledgements

NICTA is funded by the Australian Government as represented by the Department of Broadband, Communications and the Digital Economy and the Australian Research Council through the ICT Centre of Excellence program

References

[1] D. Jiang *et al.*, "Design of 5.9 GHz DSRC-based vehicular safety communication," *IEEE Wireless Commun.*, vol. 13, no. 5, pp. 36–43, October 2006.

[2] Q. Tse and B. Landfeldt, "Interference-aware geocasting for vanet," in *Proc. 13th IEEE Int. Symp. World of Wireless, Mobile and Multimedia Networks*, 2012, pp. 1–6.

[3] G. Bianchi, "Performance analysis of the IEEE 802.11 distributed coordination function," *IEEE J. Select. Areas Commun.*, vol. 18, no. 3, pp. 535–547, Mar 2000.

[4] G. Bianchi and I. Tinnirello, "Kalman filter estimation of the number of competing terminals in an IEEE 802.11 network," in *Proc. 22nd IEEE Int. Conf. Comput. Commun.*, vol. 2, March-3 April 2003, pp. 844–852.

[5] L. Bononi, M. Conti, and L. Donatiello, "Design and performance evaluation of a Distributed Contention Control (DCC) mechanism for IEEE 802.11 wireless local area networks," in *Proc. 1st IEEE Int. Symp. World of Wireless, Mobile and Multimedia Networks*. New York, NY, USA: ACM, 1998, pp. 59–67.

[6] A. Toledo, T. Vercauteren, and X. Wang, "Adaptive optimization of IEEE 802.11 DCF based on Bayesian estimation of the number of competing terminals," *IEEE Trans. Mobile Computing*, vol. 5, no. 9, pp. 1283–1296, Sept. 2006.

[7] J.-S. Kim, E. Serpedin, and D.-R. Shin, "Improved particle filtering-based estimation of the number of competing stations in IEEE 802.11 networks," *IEEE Signal Process. Lett.*, vol. 15, pp. 87–90, 2008.

[8] T. Vercauteren, A. L. Toledo, and X. Wang, "Batch and sequential Bayesian estimators of the number of active terminals in an IEEE 802.11 network," *IEEE Trans. Signal Process.*, vol. 55, no. 2, pp. 437–450, Feb. 2007.

[9] C. M. Bishop *et al.*, Eds., *Pattern recognition and machine learning*. Springer, New York, 2006.

[10] S. Bastani, B. Landfeldt, and L. Libman, "On the reliability of safety message broadcastin urban vehicular ad hoc networks," in *Proc. 14th ACM/IEEE Int. Conf. Modeling Anal. and Simulation Wireless and Mobile Syst.* New York, NY, USA: ACM, 2011, pp. 307–316.