Abstract—The existence of hidden and exposed nodes can have a significant negative impact on the performance of IEEE802.11 networks. Such nodes can increase the probability of collisions and limit the spatial reuse on the channel. The value of the Physical Carrier Sensing Threshold (PCST) is instrumental in the presence of such nodes in a system. This paper presents a new adaptive physical carrier sensing management algorithm for wireless networks. Our KAPCS2 management system maximizes the aggregate system throughput by identifying and addressing problem nodes. It incorporates 802.11k Radio Resource Measurements (RRM) on each node to enable on-line tuning of the PCST.

Keywords—Performance optimization, Algorithm design, Simulation

I. INTRODUCTION

The IEEE 802.11 standard defines two modes of operation, Distributed Coordination Function (DCF) and Point Coordination Function (PCF) [11]. However, due to the complexity of PCF and the cost of implementing it, few implementers have chosen to realize it. DCF was designed mainly for asynchronous data transport and is based on the Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) protocol. This protocol dictates that all stations must contend to gain access to the medium, where all have equal chance of accessing the channel. CSMA/CA uses carrier sensing to determine whether the medium is available by sampling the energy currently on the channel.

The carrier sensing technique employed is limited and suffers from the well documented problems of hidden and exposed nodes. Nodes are labelled as such when they are out of range of other nodes in the system or when they are wrongly prevented from transmitting to other nodes due to a neighbouring transmitter. These events result in low spatial reuse and high collision rates.

Various approaches have been made to improve the carrier sensing mechanism. Results have shown that efficient management of the PCST can significantly enhance the performance of a wireless system. The PCST, used to determine the availability of the channel, can be tuned to an optimal value with regards to maximizing system throughput.

The 802.11k standard provides mechanisms for radio resource measurements (RRM) [1]. A framework of measurement requests and reports are defined to improve the provision of data from the Medium Access Control (MAC) and Physical (PHY) layers. This data can be used in the higher layers to enable nodes to measure information that may help to optimize use of the radio network.

The work in this paper is motivated by [2], which details an IEEE802.11k enabled adaptive carrier sense mechanism for wireless networks (K-APCS). Although results indicated a significant performance gain, K-APCS also has some limitations regarding convergence and fairness. We have addressed these issues with our current adaptive carrier sense management mechanism, KAPCS2. KAPCS2 employs IEEE802.11k RRM to perform measurements on each node in the system, and to inform other nodes about their performance. This additional information enables nodes to manage and tune the PCST with the goal of maximizing throughput.

The remainder of this paper is structured as follows: section 2 contains an overview of the related work. Section 3 details the problem definition. Section 4 presents the KAPCS2 algorithm. Section 5 details the simulation scenario configured for the tests. Section 6 presents the results and discussion. Finally, section 7 concludes the paper.

II. RELATED WORK

A considerable amount of work has been carried out on the carrier sense mechanisms used in wireless networks. Zhu et al. [3] investigate the optimal PCST for various configurations of ad hoc networks. They determine that the threshold can be tuned to an optimal value with respect to maximizing the system throughput. Yang et al. [4] and Deng et al. [5] echo these findings. Zhu et al. [6] determined that tuning the carrier sense threshold to cover the entire interference range can increase the spatial reuse.

Ma et al. [7] introduce a stochastic model for optimizing physical carrier sensing and spatial reuse in wireless ad hoc networks. The motivation for their model was the Markov model developed for optimal transmission range in multihop wireless networks [6]. Their new Markov model captures the effect of PCST choice on the aggregate one-hop throughput.

Zhu et al. [8] use the model in [7] as a basis for their QoS aware aggressive physical carrier sense tuning algorithm. An online tuning mechanism was added to track the statistical
changes in the network and to tune the PCST to reflect these changes.

Our previous work in [2] includes a comprehensive comparison of K-APCS and related work in the area. Tests were carried out on five different variants of a carrier sense mechanism:

- The 802.11 standard carrier sensing mechanism
- The carrier sensing mechanism modeled in [8], bounded by the packet loss rate
- The carrier sensing mechanism modeled in [8], bounded by the frame loss rate
- K-APCS bounded by the packet loss
- K-APCS bounded by the frame loss

The results of these tests are detailed with focus placed on performance characteristics of the network such as throughput, retransmission rate, and packet loss rate. Some additional discussion is presented on the carrier sense threshold values and fairness of the system.

III. PROBLEM DEFINITION

A. System Propagation Model

The system pathloss model uses free space pathloss for the direct line-of-sight propagation path and the reflection from flat earth. Using this model the average received signal strength can be expressed as a function of the distance between the transmitter and receiver. The path loss coefficient \( \delta \) takes a value ranging from 2 to 4 depending on the environment. RSS denotes the signal strength at the receiving node \( n_{rx} \); the distance between the transmitting node \( n_{tx} \) and \( n_{rx} \) is denoted by \( d_{txrx} \). RSS is the power received at a reference point distance \( d \) from the transmitter, a typical value for this reference distance is 1 meter.

\[
\text{RSS} = \text{RSS} \left( \frac{d}{d_{txrx}} \right) ^\delta \tag{1}
\]

The total received power at a node consists of the signal, interference and noise. The receiver \( n_{rx} \) can only successfully receive the signal if \( \text{RSS} \) is greater than the PCST, denoted \( T_{pcs} \). The carrier sense range \( D_{cs} \) and the reception range \( D_{rx} \) can be calculated by (2), where \( T_{pcs} \) and \( T_{rx} \) denote the carrier sense and the receive thresholds respectively.

\[
D_{cs} = d \left( \frac{\text{RSS}}{T_{pcs}} \right) ^{\frac{1}{\delta}} \tag{2}
\]

\[
D_{rx} = d \left( \frac{\text{RSS}}{T_{rx}} \right) ^{\frac{1}{\delta}}
\]

The maximum distance at which \( n_{rx} \) can be affected by any other \( n_{tx} \) is described as the interference range of \( n_{rx} \), denoted \( D_{txrx} \) and can be calculated by (3):

\[
D_{txrx} = d_{txrx} \alpha ^{\frac{1}{\delta}} \tag{3}
\]

where \( \alpha \) is the required signal to interference noise ratio needed to decode the signal at the receiver.

Fig. 1 illustrates the relationship between the ranges detailed in (2) and (3). The interference range is depicted by the circle with the solid circumference, and the carrier sense range is represented by the circle with the dashed circumference. The shaded area is the hidden region. \( n_{h} \) is within the interference range of \( n_{rx} \) but outside the carrier sense range of \( n_{tx} \) and therefore is a hidden node. Similarly, \( n_{e} \) is within the carrier sense range of \( n_{tx} \) but outside the interference range of \( n_{rx} \) and therefore is an exposed node.

The presence of hidden nodes in a WLAN can have a significant negative effect on the performance of the system by causing a high collision rate; this is reflected in the results reported by Borgo et al. [9] and Jayasuriya et al. [10]. The 802.11 standard defines a Virtual Carrier Sensing (VCS) mechanism which attempts to address the hidden node problem [11]. Work in [12, 13, 5] suggests that VCS is not an optimal solution, and can even introduce a different yet equally detrimental problem for system performance - the presence of exposed terminals can significantly reduce the throughput of nodes in the network.

IV. PROPOSED SOLUTION

In order to improve the throughput of the nodes in the system the carrier sense threshold must be set to a value which provides a good balance between exposed and hidden terminals. The work in this paper aims to tune the physical carrier sense threshold using RRM with the goal of maximizing system throughput.

A. 802.11k Measurements

The 802.11k standard provides mechanisms for radio resource measurements. A node may choose to make MAC or PHY layer measurements locally, request a measurement from another node, or may be requested by another node to make one or more measurements and return the results.

For our work, 802.11k measurements are used to record the time the channel is busy due to the transmissions of all nodes in the carrier sensing region of the measuring station, and the time the channel is busy due to the transmissions of the measuring station only. Both measurements are performed by every node in the Basic Service Set (BSS) at predefined intervals. The channel busy measurement is used locally,
whereas the transmission time is reported to the Access Point (AP).

We introduced a new measurement report called the **activity report**; this is used to send the transmission time recorded by nodes in the BSS. Each node periodically sends an activity report to the AP. The AP compiles a list of each node in the BSS with the corresponding transmission time. This information is updated every interval and broadcast in the beacon frame.

1) **Tracking Dynamic Conditions**

The CSMA mechanism employed by 802.11 devices enables each user to sense all transmissions which occur in its PCS range. In this way, each user can maintain statistical information on-line. The three dynamic network statistics of interest are:

- The time the channel is busy due to the transmissions of nodes in the carrier sense range, denoted \( T_{\text{cs\_range}} \)
- The time the channel is busy due to transmissions from the node itself, denoted \( T_{\text{tx}} \).
- The retransmission rate of the node, denoted \( R_{\text{tx}} \).

We update these values at intervals of 500ms.

During interval \( i \) \((T_{\text{cs\_range}}^{(i)}, T_{\text{tx}}^{(i)}, R_{\text{tx}}^{(i)})\) are recorded at the MAC layer. After the \( i \)th interval these values are updated by

\[
T_{\text{cs\_range}}^{(i+1)} = \lambda T_{\text{cs\_range}}^{(i+1)} + (1 - \lambda) T_{\text{cs\_range}}^{(i)}
\]

\[
T_{\text{tx}}^{(i+1)} = \lambda T_{\text{tx}}^{(i+1)} + (1 - \lambda) T_{\text{tx}}^{(i)}
\]

\[
R_{\text{tx}}^{(i+1)} = \lambda R_{\text{tx}}^{(i+1)} + (1 - \lambda) R_{\text{tx}}^{(i)}
\]

where \( \lambda \) is the smoothing factor with the value \( \lambda = 0.9 \). This value was used as extensive testing determined that it provides a good compromise between accuracy and promptness.

2) **Activity Report**

The Activity Report returns a picture of the traffic load generated by the reporting node. It contains the \( T_{\text{tx}} \) statistic measured locally and the ID of the node. The format of the Measurement Report field corresponding to an Activity Report is shown in Fig. 2. The first 12 octets are defined by the IEEE802.11k standard.

<table>
<thead>
<tr>
<th>Regulatory Class</th>
<th>Channel Number</th>
<th>Actual Measurement Start Time</th>
<th>Measurement Duration</th>
<th>NodeID</th>
<th>Tx Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Octets: 1</td>
<td>1</td>
<td>8</td>
<td>2</td>
<td>6</td>
<td>1</td>
</tr>
</tbody>
</table>

![Activity Report Frame](image)

Regulatory Class indicates the channel set for which the measurement request applies. Country, Regulatory Class, and Channel Number together specify the channel frequency and spacing for which the measurement request applies.

Channel Number indicates the channel number for which the measurement report applies. Channel Number is defined within a Regulatory Class.

Actual Measurement Start Time is set to the value of the measuring nodes’s timer at the time the measurement started. Measurement Duration is set to the duration over which the Activity Report was measured.

NodeID is the ID of the reporting node. Tx Time contains the proportion of measurement duration for which the measuring node determined the channel to be busy due to its own transmissions.

3) **Modifications to Beacon Frame**

The AP maintains a Station Activity List (SAL) of nodes in the BSS and their corresponding \( T_{\text{tx}} \) statistic. On receipt of an Activity Report the AP updates the reporting node’s SAL entry. Nodes associated with the AP are denoted \( \text{Node}_{\text{bss}} \). For ease of implementation, the SAL is currently added to the beacon frame but could be sent in a dedicated report frame.

Each node maintains a Neighbours List (NL), this is the set of IDs identifying nodes in the reception range. This list is cleared after every interval and is repopulated on receipt of frames from nodes within the reception range, denoted \( \text{Node}_{\text{rx}} \).

B. **Characterizing and Identifying Problem Nodes**

A problem node is defined as a node which is either hidden or exposed. Problem nodes can be identified by comparing the list of nodes the station can see in the reception range, to the list of associated nodes reported by the AP. The set, \( A \), of \( \text{Node}_{\text{rx}} \) and the set, \( B \), of \( \text{Node}_{\text{bss}} \) can be defined by (5) and (6) respectively.

\[
A = \{ x | x \text{ is in NL}, \text{Node}_{\text{rx}} \in A \} \quad (5)
\]

\[
B = \{ x | x \text{ is in SAL}, \text{Node}_{\text{bss}} \in B \} \quad (6)
\]

We can define the set, \( C \), of nodes outside the reception range \( \text{Node}_{\text{cs\_h}} \) by (7). These nodes may be in the carrier sense range or the hidden region.

\[
C = \{ x | x \in B \setminus A \}, \text{Node}_{\text{cs\_h}} \in C \quad (7)
\]

The total time the channel was busy due to the transmissions of nodes outside the reception range, \( T_{\text{cs\_h}} \) can be calculated by (8).

\[
T_{\text{cs\_h}} = \sum_{n=0}^{n=|C|} \text{Node}_{\text{cs\_h}} \quad (8)
\]

Hidden nodes \( \text{Node}_{\text{h}} \) can be identified by (9).

\[
\text{Node}_{\text{h}} = (C \neq \emptyset) \land (T_{\text{cs\_range}} < T_{\text{cs\_h}}) \land (\exists \text{Node}_{\text{cs\_h}} \in C \left( \text{Node}_{\text{cs\_h}} \leq |T_{\text{cs\_h}} - T_{\text{cs\_range}}| \right) \quad (9)
\]

The set, \( D \), of nodes which are exposed to a station within its reception range, denoted \( \text{Node}_{\text{rx\_e}} \) can be defined by (10).
Exposed nodes $N_e$ can be identified by (11).

$$N_e = (D \neq \emptyset) \lor (T_{cs,range} > T_{cs,h})$$

C. Heuristic Carrier Sense Threshold Tuning Algorithm

The KAPCS2 algorithm tunes the carrier sense threshold $T_{pcs}$ at intervals using the steps detailed below. The presence of a hidden node indicates that the current $T_{pcs}$ is not sufficiently sensitive. KAPCS2 addresses this problem by decrementing the $T_{pcs}$. This is repeated until the node can sense everyone in the BSS. Similarly, the presence of an exposed node indicates that the current $T_{pcs}$ is too sensitive, KAPCS2 addresses this problem by incrementing the $T_{pcs}$ in intervals until the node can only sense the transmissions of nodes in the BSS.

```
Initialise $T_{pcs}$
Do
    Obtain $\{ T_{cs,range}, T_{tx}, Re_{tx} \}$ statistics
    Update $\{ T_{cs,range}, T_{tx}, Re_{tx} \}$ by (4)
    Update NL
    Tx.Activity Report by IEEE802.11k RRM
    Broadcast SAL in Beacon Frame
    Calculate $T_{cs,h}$ by (8)
    Identify $N_h$ and $N_e$ by (9) and (11)
    Update $T_{pcs}$ by $if(N_h) T_{pcs} -= 1$
    $if(N_e) T_{pcs} += 1$
End do
```

Revisiting Fig. 1, we consider a scenario where $D_{cs}$ is small enough that it does not include $n_e$, but does include $n_h$. In this case, KAPCS2 will increase the $D_{cs}$ (decrease the $T_{pcs}$) to try to eliminate the hidden node. $D_{cs}$ may eventually include both $n_e$ and $n_h$. The introduction of $n_e$ would cause an increase in $T_{cs,range}$ and could lead to the mechanism wrongly concluding that the hidden node was eliminated.

KAPCS2 addresses this issue by maintaining a record of the $Re_{tx}$ on each node. Eliminating a hidden node will decrease the $Re_{tx}$. When an increase in $T_{cs,range}$ has been achieved, KAPCS2 compares $Re_{tx}^{(i)}$ and $Re_{tx}^{(i+1)}$ to determine if the hidden node has been eliminated. If $Re_{tx}$ has not improved, KAPCS2 will conclude that the node is now suffering from both a hidden and an exposed problem. KAPCS2 will increment the $T_{pcs}$ and suspend tuning on that node for 30 seconds. Similar behavior is displayed when the opposite occurs, i.e. when $D_{cs}$ is decreased and $n_h$ is introduced.

In the event of a node being both hidden and exposed there are various issues to consider. When $D_{cs}$ includes $n_h$, both nodes belong to the BSS and the system suffers from two problem nodes. When $D_{cs}$ includes $n_e$, possibly only one node in each BSS are affected. Both events have a significant negative impact on system performance. KAPCS2 does not consider this scenario at present, more investigation is required to correctly evaluate which problem is most beneficial to address.

V. SCENARIO DESIGN

Extensive tests were carried out using the Qualnet Simulation tool. Each simulation was run for 100s. Saturated nodes send CBR packets to a wired server. Each node uses different power and data rate depending on its distance from the AP. Each node starts an application at a random time between 1s and 10s. The KAPCS2 mechanism is started at 30s.

<table>
<thead>
<tr>
<th>Simulation Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>Propagation Channel Frequency</td>
</tr>
<tr>
<td>Propagation Pathloss Model</td>
</tr>
<tr>
<td>Propagation Shadowing Model</td>
</tr>
<tr>
<td>Propagation Shadowing Mean</td>
</tr>
<tr>
<td>PHY Model</td>
</tr>
<tr>
<td>Antenna Model</td>
</tr>
<tr>
<td>Antenna Height</td>
</tr>
</tbody>
</table>

Fig. 3 - 6 illustrate the network topologies used in the simulation scenarios for this research. All APs are operating on the same channel. These topologies could, for example, represent a scenario which may exist in an apartment block or building comprised of small offices. Residents may simply set up a wireless AP and proceed to use the default channel, with no consideration of network planning.

Four different topologies were designed to test the KAPCS2 mechanism. Scenario 1 consists of two subnets, with one exposed node in each, see Fig. 3. This is the simplest form in which a wireless network can suffer from the exposed node problem.

Scenario 2 contains one subnet with two hidden nodes, see Fig. 4. This is the simplest form in which a wireless network can suffer from the hidden node problem.

Figure 3. Scenario 1 – Exposed

Figure 4. Scenario 2 – Hidden
The third scenario is made up of three subnets with both hidden and exposed nodes, see Fig 5. The hidden nodes are depicted by the squares and the exposed nodes by the circles.

Scenario 4 uses the same topology as scenario 3 but is more dynamic in nature. Mobility and dynamic network associations were introduced to increase the complexity of the scenario, thus increasing the realism of the test. Stations A and B are mobile nodes, this mobility causes the problem status of the nodes to change dynamically during the simulation. Station C joins the network at 50s and introduces an additional hidden node into the system. This test allows us to evaluate how KAPCS2 handles a non-static network scenario.

VI. SIMULATION RESULTS

A. Physical Carrier Sense Threshold

Fig. 7 plots the PCST for scenario 2. Various tracking intervals were tested. Results show that the convergence speed of the algorithm for a static environment is dependent on the interval used. In scenario 2, KAPCS2 converges in 1.5s and 3s, when interval $i = 500$ms, and 1s respectively. Oscillation occurs when the smaller intervals of 10ms and 100ms are used. The value $i = 500$ms was chosen for the remainder of the tests. Smaller values of $i$ resulted in greater oscillation when the density of nodes in the system increased.

Fig. 8 plots the PCST results for scenario 1 and 2. The results show that exposed nodes E1 and E2 decreased their sensitivity by increasing the PCST from -85 to -80. At these new values, they can no longer sense the other’s transmissions and are no longer exposed to each other.

The hidden pair H1 and H2 increased their sensitivity by decreasing the PCST from -85 to -87 respectively. At this stage they begin to sense the other’s transmissions on the channel and are no longer hidden from each other. The system converges in 2.5s.
Fig 9 plots the PCST results for scenario 4. The mobile node A moves 195m in a SW direction and then returns to its original position. As A moves closer to B, the graph shows that KAPCS2 increases the PCST to try to eliminate the exposed node B. As A continues to move closer to B the PCST is increased again. When A begins to move away from B and back to its original position the PCST value remains static. KAPCS2 will only decrease the threshold if a hidden node is detected.

The mobile node B moves 100m in an N direction. When KAPCS2 is started at 30s it detects that B is hidden from H2. The PCST is decreased to increase sensitivity and to eliminate the hidden node. As B continues to move north, it gets closer to the hidden node H2 and the exposed node E1. KAPCS2 detects that B is no longer suffering from a hidden node problem. It also detects the newly introduced exposed node problem. The PCST is increased to decrease sensitivity, thus eliminating the exposed node.

The static node C joins the network at 50s, this association introduces a hidden node problem into the system. The results show that KAPCS2 decreases the PCST of node C at 50s. The PCST of the other hidden node, H3, is also decreased.

B. Throughput

Fig. 10 plots the throughput results for scenario 1 and 2. Tests were carried out on both the IEEE802.11 and KAPCS2. Results show that KAPCS2 achieves an average of 100% throughput gain for exposed nodes in scenario 1 and 100% throughput gain for hidden nodes in scenario 2. For scenario 3, shown in Fig. 11, KAPCS2 achieves an average 38% gain in system throughput. The elimination of problem nodes decreases the collision rate and increases the spatial reuse, thus enabling increased performance. Similar performance gain can be seen for scenario 4.

C. Frame Loss

Fig. 12 graphs the frame loss results for all scenarios. KAPCS2 addressed the hidden node problem; hidden pairs become more sensitive, and back off when necessary. This reduces the collision rate in the system by an average of 83% for scenario2. The results show a similar behaviour in scenario 3, KAPCS2 eliminates the hidden nodes in the system and achieves an average 23% decrease in frame loss.

D. Fairness

The unique characteristics of the wireless channel, such as location-dependent contention and the absence of any centralized control leads to difficulty in solving fairness issues.
The work in this paper aims to maximize spatial reuse to increase the utilization of the wireless channel. In doing this, we encounter a fundamental conflict between optimizing the throughput and achieving fairness. Allocating a channel to a flow with a large contention would reduce the channel reuse and the corresponding throughput.

![Figure 13. Mean Packet Tx](image)

Fig. 13 charts the standard deviation in the mean number of unicasts transmitted in the system for the K-APCS mechanism detailed in [2] and the KAPCS2 mechanism. All mechanisms suffer from fairness issues; however, there is an increase in the standard deviation for both of the K-APCS mechanism variants. We have addressed this problem with our current mechanism. KAPCS2 does not increase the level of unfairness seen with the traditional 802.11 carrier sensing mechanism, while still affording a significant increase in channel utilisation and decrease in collision rates.

VII. CONCLUSION

The objective of this paper was to maximize aggregate system throughput by the efficient management of physical carrier sensing. Optimizing the PCST value enables increased channel utilization with acceptable collision rates through the elimination of trade-off between the existence of hidden and exposed nodes.

Our KAPCS2 management algorithm was implemented in QualNet and compared to the 802.11 mechanism using various network scenarios. Network statistics such as throughput, PCST, and retransmission rate were recorded to gauge system performance. Results show that the KAPCS2 mechanism affords a significant performance gain: it achieved at least 38% increase in throughput, and at least 23% decrease in frame loss. These results indicate that KAPCS2 is an effective mechanism for tuning the carrier sense threshold.

ACKNOWLEDGMENT

The research leading to these results has received funding from the European Community’s Seventh Framework Programme (FP7/2007-2013) under grant agreement nº 214994.

This research is partially funded by Science Foundation Ireland (SFI) via grant 08/SRC/11403 FAME SRC (Federated, Autonomic Management of End-to-End Communications Services - Scientific Research Cluster)

REFERENCES

[12] K. Xu, M. Gerla, S. Bae, “How effective is the IEEE 802.11RTS/CTS handshake in ad hoc networks?” GLOBECOM