Enhanced IEEE 802.11 by Integrating Multiuser Dynamic OFDMA

Hasan Shahid Ferdous and Manzur Murshed, *Member*, *IEEE*
Faculty of IT, Monash University, Australia
Email: {Ferdous.Hasan, Manzur.Murshed}@infotech.monash.edu.au

***Abstract*—In this paper, we discuss the problems associated with the present contention resolution mechanism of IEEE 802.11 DCF and present a new, dynamic and robust approach to improve it. Our new MAC, using OFDMA in the physical layer, can incorporate multiple concurrent transmissions or receptions in a dynamic manner and can adjust the collision probability based on the traffic load when nodes are endowed with a single half-duplex radio only. Simulation results show that our system improves throughput by up to 40 percent, reduces collision in control messages by up to 80 percent and reduces the average delay for data transmission by up to 20 percent.**

I. Introduction

Wireless computing devices play a major role in today's world. New wireless technologies like wireless regional area networks are emerging while use of the existing technologies like wireless local area networks (WLAN) and wireless personal area networks are increasing steadily. The issue of improving their performance is thus becoming more and more important everyday. Since its inception in 1997, IEEE 802.11 has seen several modifications and almost omnipresent today. Technology used in its physical layer (PHY) has been changed significantly. At first, frequency hopping spread spectrum and infrared were used. Then came direct sequence spread spectrum, and finally, orthogonal frequency division multiplexing (OFDM) is established as the principle PHY for 802.11. All the latest standards of WLAN like IEEE 802.11 a/g/y/n recommend OFDM as the PHY technology. The 802.11 working group has started discussions on future WLAN advancement based on two different approaches - utilizing the 60 GHz frequency band [1], and increasing the aggregated throughput by exploiting multiuser diversity [2], [3]. OFDM will remain as the basic prevailing transmission scheme for both of these projects.

Multiple input multiple output (MIMO) can incorporate concurrency by enabling nodes to receive n data streams in parallel [4] or eliminate interference from as many as $(n-1)$ neighbors [5]. But it requires the nodes to have at least n antennas and consumes higher energy. Another approach to obtain concurrency is to use OFDMA (Orthogonal Frequency Division Multiple Access). In OFDM systems, subcarriers or tones are orthogonal carriers of lower-rate input data streams. It results in longer symbol duration compared to channel delay spread to mitigate multipath effects. In OFDMA, a group of non-overlapping subcarriers, called sub-channels, are assigned to each user to enable simultaneous multiple data transmissions to a base station or vice versa. Preserving the benefits of OFDM (frequency diversity, spectral efficiency, and low inter symbol interference), it can enhance the efficiency in the PHY layer by increasing multiuser diversity and decreasing peak to average power ratio (PAPR), which is a major drawback in designing OFDM based hardware. Considering the benefits of OFDMA, WiMAX is using it from the very beginning. IEEE 802.11 can also be benefitted from it. Using OFDMA, we can incorporate multiple concurrent transmissions or receptions in IEEE 802.11 and reduce the drawbacks of it to a significant extent.

There are many works on synchronization [6], spectrum allocation [7] and signalling [8] in OFDMA based systems. But few works discussed about implementing OFDMA in IEEE 802.11. Valentine *et al.* [9] discussed the issues that are needed to be considered to implement OFDMA in IEEE 802.11. Another paper of them [10] described a practical implementation of their proposed system in a real software radio test-bed and proved the effectiveness of their design. They discussed about the medium access control (MAC) and logical link control (LLC) layer extensions required for implementing OFDMA, a fair and optimized policy for physical protocol data unit (PPDU) fragmentation, and sub-channel allocation by using their extended request to send (eRTS) and extended clear to send (eCTS) control messages. The main drawback of their work is that they considered only downlink traffic, i.e. data is sent only from the access point to the wireless stations. We will preserve most part of their work and extend it to support uplink.

Veyseh *et al.* [11] proposed their distributed coordinating function (DCF) named CTRMA (Concurrent Transmission or Reception Multiple Access) and compared their work with single-radio single-channel, single-radio multi-channel, and multi-radio multi-channel MAC protocols and found the superiority of their proposal in network throughput. The main drawbacks of their CTRMA protocol is that, it requires extensive message exchange between the users, in O(d^2C) where d is the average number of neighbors and C is the number of channels; and it is required every time the network topology changes. They showed very little experimental results and their performance comparison is based on network throughput only. They did not discuss about the average delay or other performance matrices of their system. They assumed separate

978-1-4244-6557-6/10/$26.00 ©2010 IEEE

control channels for RTS/CTS exchange and information negotiation, which decreases the data bandwidth. These control channels are narrower than the data channel. So, probability of congestion and collisions in these channels may increase.

In another paper, Veyseh *et al.*[12] proposed another new DCF based on OFDMA that incorporates multiple concurrent transmissions or receptions. But this proposal is based on a receiver initiated collision avoidance protocol, which is not used in traditional wireless devices.

In our work, we will show the inherent reasons of the inefficiency of present 802.11 DCF and the motivations of this research. Then, we will provide a solution that reduces these problems and improves the system's performance. We will show that the loss due to the four way handshaking mechanism in IEEE 802.11 has significant impact on the achievable throughput and the collision probability increases rapidly with the increase of the number of nodes present in the system. Then we will propose a solution that can incorporate multiple concurrent transmissions or receptions using OFDMA and thus decreases collisions, increases throughput and reduces the average delay.

We organize our work into the following sections. Section II discusses about the present 802.11 DCF and its drawbacks. In Section III, we present our approach and discuss about how we can overcome the drawbacks of 802.11 and in Section IV, we discuss about our simulation setup and results. Finally, in the concluding section, we consolidate our work and state our future research directions in this area.

II. Present IEEE 802.11 DCF

IEEE 802.11 was designed as *Ethernet in the sky*. Since its inception, tremendous research efforts are deployed to improve its efficiency. There are numerous papers like [13] that analyzed the performance of IEEE 802.11 and discussed about the ways to improve it. IEEE 802.11 has two modes of operation - DCF (mandatory) and PCF (Optional). To understand our work, we need to discuss the 802.11 DCF briefly (see Fig. 1).

When a node wants to transmit a packet, it waits until the channel is sensed idle for an interval named distributed inter frame spacing (DIFS). If the channel is sensed busy during the DIFS period, the node persists to monitor the channel until it is measured idle for a DIFS time. At this point, the node chooses a random backoff interval before transmitting to minimize the probability of collision with other nodes. If any other node captures the channel during this backoff period, it waits for the channel to remain free for another DIFS period. Otherwise, it transmits a request to send (RTS) message. The receiving node responds, after a SIFS (Short Inter Frame Spacing) time, with a clear to send (CTS) frame. The RTS and CTS frames carry the information about the length of the packet to be transmitted. This information can be read by any listening node, which is then able to update a network allocation vector (NAV) to estimate the period of time in which the channel will remain busy. Therefore, collision can occur only when two node chooses the same backoff slot to send RTS. If a

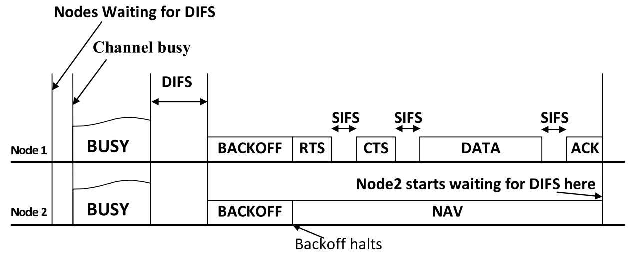

Fig. 1: IEEE 802.11 DCF.

collision is detected (by not receiving CTS), the node doubles its backoff window unless it reaches its maximum limit. Then it waits for the channel to remain free for EIFS (Extended Inter Frame Spacing) time. When a node receives data correctly, it sends an acknowledgment (ACK) message to the sender. We will see the impact of this four way handshaking on the performance of IEEE 802.11 protocol.

In [14], Xiao *et al.* analyzed the performance of IEEE 802.11 and formulated a mathematical limit for the practical throughput of it. He proved that there is a throughput upper limit that depends on the packet size of the transmission. His formula matches almost perfectly with the practical throughput analysis of IEEE 802.11. Practical data throughput is almost 50% of the theoretical capacity of the network.

We can understand the reasons behind this inefficiency of IEEE 802.11. Wireless channel is inherently broadcast in nature. Nodes are half-duplex, so they need RTS/CTS mechanism to avoid the hidden node problem. Again, all the control messages like RTS, CTS and ACK are sent at the lowest speed (6 Mbps) while data can be sent at a maximum speed of 54 Mbps (802.11a). So, these messages occupy a significant portion of the transmission time available. And according to our analysis shown in the next section, RTS collision probability is not negligible, even for moderate number of nodes. Time is also wasted during backoff, DIFS and SIFS intervals. Collisions increase the wasted time of backoff too. For every successful packet transmission, 123 μs is wasted to exchange these control messages and inter frame spacing times, in addition to the backoff period. For every RTS collision, another 84 μs is lost. We will discuss a simple mechanism in the next section to decease this wasted time and thus improve the performance of IEEE 802.11.

III. Our Proposal and System Description

A. Main Idea

The basic idea of our proposed DCF has its origin in the famous *Birthday Problem*. The birthday problem pertains to the probability that in a set of randomly chosen people, some pair of them will have the same birthday. Assuming that the 365 possible birthdays are equally likely, it can be shown that the probability of having at least one pair with the same birthday is close to 1 with as few as 57 people. Similarly, for any contention window of size s, the probability of having two or more nodes selecting the same slot(s) will be significantly high even when the number of nodes n is much less than s.

While due to exponential backoff in IEEE 802.11 DCF, contention windows size among the nodes may vary at any instance, for the sake of simplicity, let us assume that the same s is being used by all the contending nodes. The probability of having exactly k collisions by the n contending nodes can be calculated as

$$P_e(n,k) = \frac{1}{s^n}\begin{cases} P_n^s, & k=0; \\ \sum_{i_1=2}^{n-2}\cdots\sum_{i_k=2}^{n-\sum_{j=1}^{k-1} i_j}\binom{s}{k}\binom{n}{i_1}\cdots\binom{n-\sum_{j=1}^{k-1} i_j}{i_k} P^{s-k}_{n-\sum_{j=1}^{k} i_j}, & k>0, \end{cases} \quad (1)$$

where $P_r^n = n(n-1)\dots(n-r+1)$. The probability of having k or more collisions can then be derived as

$$P(n,k) = 1 - \sum_{i=0}^{k-1} P_e(n,i). \quad (2)$$

We have plotted $P(n,k)$ in Fig. 2 for $k = 0, 1, \dots, 4$ with $s = 32$. Clearly, the probability of collisions increases sharply with the number of contending nodes. Even more concerning is that the probability of multiple collisions also increases very rapidly. As both throughput and delay of the network are affected by the collisions, 802.11 DCF can no longer be deemed efficient.

Dividing the subcarriers of an OFDM PHY into sub-channels to effectively allow disjoint groups of nodes to contend using individual sub-channels may improve the efficiency, if we can show that the overall collision probability decreases. Let us assume that n nodes are divided into c groups such that the i-th group has $n_i = \lceil n/c \rceil$ nodes, for $i \leq n \bmod c$, or $\lfloor n/c \rfloor$ nodes, otherwise. Then the probability of having exactly k collisions is

$$P_e(n,k,c) = \Sigma_{\forall(k_1,\dots,k_c)\in\rho(k,c)} \Pi_{i=1}^{c} P_e(n_i,k_i) \quad (3)$$

where $\rho(k,c) = \{(k_1,\dots,k_c) | \sum_{i=1}^{c} k_i = k \bigwedge k_i \geq 0\}$. The probability of having k or more collisions can then be derived as

$$P(n,k,c) = 1 - \sum_{i=0}^{k-1} P_e(n,i,c). \quad (4)$$

From Fig. 3, which plots $P(n,k,c)$ for different values of k and c, we can conclude that

$$\forall k: P(n,k,c_i) \geq P(n,k,c_j) \text{ iff } c_i < c_j, \quad (5)$$

i.e., the probability of collision decreases significantly when nodes contend in groups using individual sub-channels. In the following section we consider the impact of sub-channelization on network throughput and delay.

B. Benefits of Sub-channelization

Sub-channelization can improve the efficiency of both the contention and transmission phases of the 802.11 DCF. In order to report the maximum achievable efficiency, let us assume that all the nodes are operating in saturation such that they always have data to transmit and the data packet size is constant.

Let c_{max} be the maximum number of equi-bandwidth sub-channels possible in the system such that each one can guarantee the minimum bandwidth requirement per node. So long $n \leq c_{max}$, we can eliminate collision altogether by dividing the subcarriers equally, as much as possible, to allow each node to operate on a separate sub-channel of almost the same bandwidth, i.e., $c = n$. In such cases, no backoff is required and only one DIFS time will be wasted for sending n RTS messages in the contention phase. Although each RTS message now takes n times longer duration, total time for sending n RTS messages remains unchanged due to parallel transmission.

When $n > c_{max}$, we cannot avoid collisions by using the maximum possible sub-channels, i.e., $c = c_{max}$ Some or all sub-channels must now be allocated to more than one nodes, requiring backoff to manage collision. As only the maximum backoff time chosen by all the nodes will be wasted, there is no extra backoff overhead due to sub-channelization. Still only one DIFS time, in addition, will be wasted. As collision

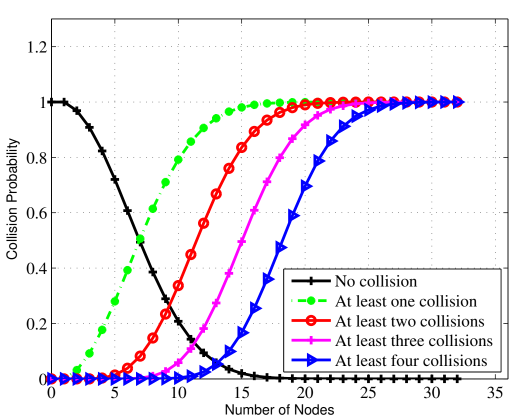

Fig. 2: Probability for different number of collisions vs. number of nodes.

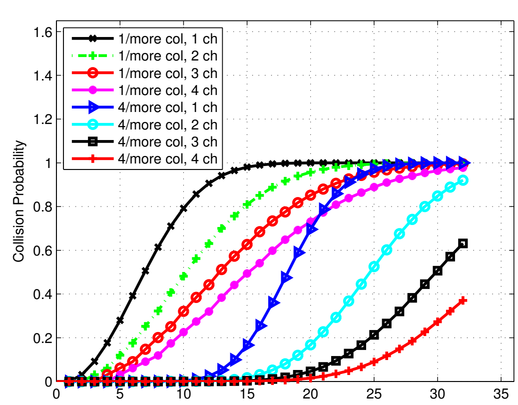

Fig. 3: Probability for different number of collisions in sub-channelized DCF vs. number of nodes.

probability reaches its minimum with c_{max} sub-channels, total wastage for collisions to send c_{max} RTS messages will be significantly reduced compared to the standard 802.11 DCF.

The contention cycle ends as soon as each group either transmitted an RTS or a collision took place. Due to collisions, number of successful RTS messages per contention cycle may not always be equal to the number of sub-channels, $c = \min(n, c_{max})$. For the sake of simplicity, let us assume that n nodes require n/c contention cycles to send one RTS message per node, i.e., $P_e(n, k, c)$ also represents the probability of having exactly k collisions in n/c contention cycles. The expected number of successful RTS messages in each contention cycle can then be approximated as

$$\bar{r} = c\left(1 - \frac{1}{n}\sum_{k=1}^{n/2} kP_e(n, k, c)\right). \quad (6)$$

At the end of the contention cycle, successful $r \leq c$ nodes take part in the transmission phase in parallel by dividing the subcarriers into r equal sub-channels, as much as possible. One consolidated CTS message can be send to save $(r-1)$ SIFS, CTS headers, and CTS error-checking codes. Similar savings can also be achieved by consolidating the ACK messages. After receiving the consolidated CTS, each of the r nodes start transmitting data simultaneously in their respective sub-channel, wasting only one SIFS time.

As time wastage is reduced significantly in both the contention and transmission phases, sub-channelization can improve network throughput beyond any doubt. We plan to report the extent of improvement through discrete event-driven simulations in Section IV. Improvement in the expected delay experienced in the network may not be as great. Sub-channelization inherently increases expected delay in packet transmission as explained below.

Let the combined transmission time for an RTS and a data packet over the whole spectrum, as done in the standard 802.11 DCF utilizing all the subcarriers in a single channel, be T. If this spectrum is divided into r sub-channels, the combined transmission time for an RTS and a data packet would be rT. Now the expected transmission delay, while transmitting r packets over the standard 802.11 network, is $(r+1)T/2$. The same under sub-channelization is rT. Consequently, the latter incurs additional $(r-1)T/2$ time in the expected transmission delay. Whether this additional delay nullifies all the gains achieved by reducing wastage time in collision avoidance, inter-frame spacing overlapping/sharing, and control signal consolidation is investigated in Section IV.

C. Proposed System

To compare our proposed system with the current IEEE 802.11a standard, we are not using special features like adaptive modulation and coding (AMC). For this paper, we consider that the nodes always have data to send to the AP (uplink). If, AP wants to send data to them (downlink), methods discussed in [9] can serve the purpose. We can describe the uplink of the system as follows:

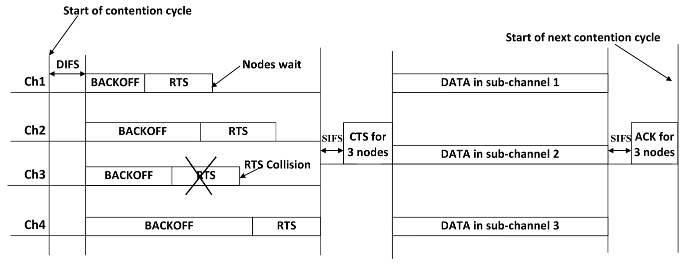

Fig. 4: The proposed DCF.

16	16	48	6	6	3	1		48	32
Frame Control	Duration	Destination Address 1	Starting Sub-carrier no	Ending Sub-carrier no	Speed	1	....	Source Address	FCS

(a) Consolidated CTS message format

16	16	48		48	48	32
Frame Control	Duration	Destination Address 1	....	Destination Address J	Source Address	FCS

(b) Consolidated ACK message format

Fig. 5: Enhanced IEEE 802.11 CTS/ACK messages for dynamic OFDMA signalling.

1) AP determines the set of associated nodes and divides them into multiple groups as discussed earlier. As the network topology changes, it broadcasts periodic message to the nodes describing the number of channels, and which node belongs to which group. Nodes in the same group share the same frequency sub-channel.
2) In every group, nodes contend among themselves to capture the channel as in IEEE 802.11. However, if only one node is assigned to a group, no backoff is required.
3) After transmitting RTS, the sender(s) waits for CTS. Other nodes in its group halt their backoff until the next contention cycle starts so that there can be no more than one successful RTS sent from one group in one contention cycle.
4) The contention cycle ends as soon as each group either transmitted an RTS or a collision took place.
5) AP calculates the subcarrier assignment using some allocation algorithm described in [7].
6) AP informs the nodes that have successfully completed RTS transmission about the assigned subcarriers using one consolidated CTS.
7) After waiting SIFS time, nodes transmit simultaneously using their assigned subcarriers.
8) Finally, after waiting SIFS time, AP transmits ACK in a single message.

We can summarize our proposed DCF in Fig. 4. The consolidated CTS and ACK message formats are shown in Fig. 5.

IV. Simulation Results

To analyze the performance of our system, we developed a discrete event driven simulation using SimJava2 [15] and examined every aspect of our system. We assumed perfect channel conditions so that we can consider only the impact of our modifications. We carefully observed the collision rate, channel utilization rate, and delay.

We tried to remain close to the IEEE 802.11a protocol. So, we set packet size = 1024 bytes, $CW_{min} = 32$, $CW_{max} = 1024$, $Retry_{max} = 7$, number of subcarriers = 48, SIFS = 4 μs, DIFS = 28 μs, EIFS = 38 μs, and slot time = 9 μs as per IEEE specifications [16]. We did not use adaptive modulation and coding; rather we used fixed data rate of 36 Mbps to simulate both traditional 802.11 and our proposed system.

We have run each of our simulations for 1000000 μs. The AP will increase the number of channels from 1 to c_{max} as n increases. We set maximum number of channels $c_{max} = 2, 4, 8, 16$ and examined how the system performs. When $n < c_{max}$, no backoff is required.

Fig. 6 shows the number of RTS messages vs. the number of nodes present in the system. In all cases, the number of RTS message increases as the number of nodes increases. But, for the same number of nodes, total number of RTS message decreases as we increase c_{max}. This is due to the fact that as c_{max} increases, less time is wasted in contention cycle, and less RTS messages are wasted in collisions, as we will see next that the total number of successful RTS message increases when we increase c_{max}.

Fig. 7 shows the number of CTS messages vs. the number of nodes present in the system. As expected, we can see that for traditional 802.11, number of CTS message grows rapidly as the number of nodes reaches up to 4 and then remains almost constant as we are considering saturation load. It slightly decreases as the number of nodes increases because the wastage due to collisions grows very rapidly. Using multiple channels can improve the output significantly. For each of the values of c_{max}, number of CTS message is very high up to $n = c_{max}$, as for these cases, there is no backoff used and collision probability is zero. Then, number of CTS message decreases to almost a fixed value due to the saturation load. As we expected, higher c_{max} values shows better results.

In Fig. 8, we considered the rate of collision in RTS messages and compared our system with the standard IEEE 802.11. Here, we can see that as the number of nodes increases, rate of collision increases rapidly in an 802.11 network. Though IEEE 802.11 tries to reduce collisions by increasing the contention window size when a collision occurs, we can see that, in a system of 15 nodes only, about 35% of the RTS messages sent were lost due to collisions. Sub-channelization can decrease collision rate significantly, as we can see that increasing c_{max} decreased collision rate. With $c_{max} = 8$, collision rate for 15 nodes is about 5% only. This is coherent with our decision based on (5).

As our fourth measure, we considered the average delay required to send a packet. We calculated delay as the time

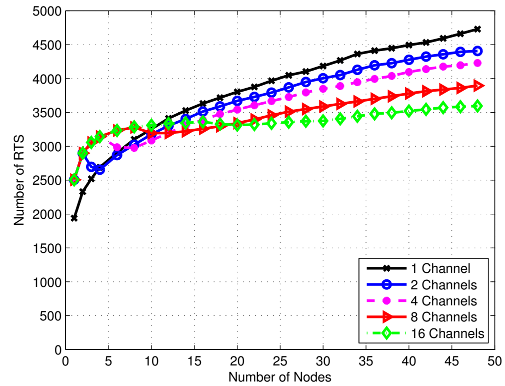

Fig. 6: Number of RTS messages for different c_{max} vs. number of nodes.

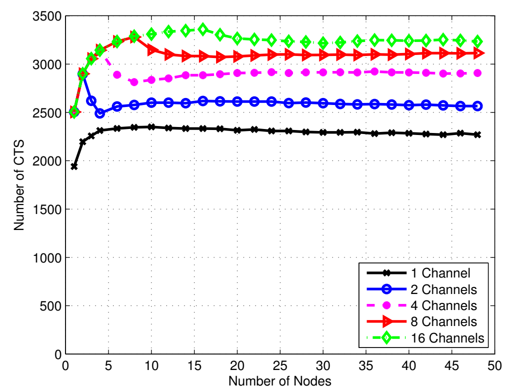

Fig. 7: Number of CTS messages for different c_{max} vs. number of nodes.

from when a node starts waiting to transmit RTS to the time it successfully completes its transmissions including retransmissions. If we use sub-channelization, we can see in our simulation that the start time of data transmission decreases significantly, but total time for transmission for each node increases, as the bandwidth is divided into multiple nodes. So, one can expect that, sub-channelization will increase the average packet transmission delay. But, in Fig. 9, we can see that sub-channelization can actually reduce this delay. For $n \le c_{max}$, the main contributing factor is that, they do not need backoff. For higher values of n, we showed earlier that wastage due to collision is no longer negligible in IEEE 802.11. Sub-channelization reduces collision and also make savings in bandwidth by reducing unnecessary inter frame spacing times and doing so, effectively reduces the average delay by up to 20%.

Then we considered the percentage of time when the channel is used for payload data. Surprisingly, we found that traditional 802.11 uses the channel to transmit payload data for only 53% of the time on an average, whereas using sub-channelization we can easily reach up to 75% of the total time (see Fig. 10). As in the case of CTS, when $n \le c_{max}$ utilization is very high. Then it decreases slightly to a almost

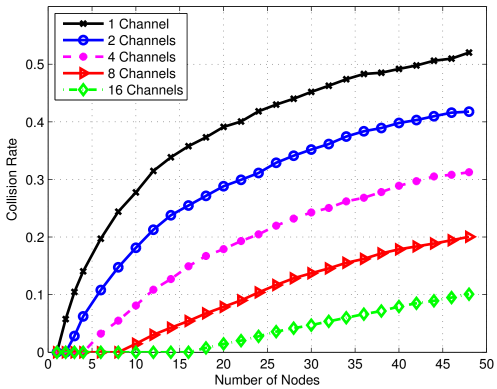

Fig. 8: RTS collision rate for different c_{max} vs. number of nodes.

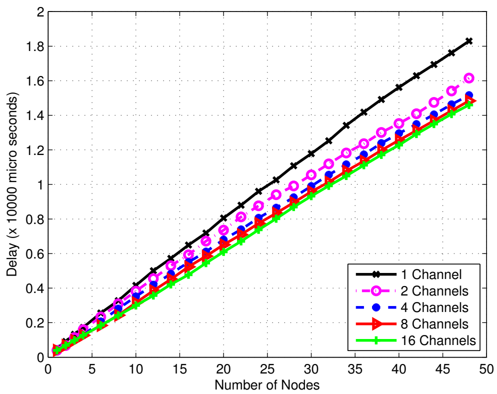

Fig. 9: Average packet delay for different c_{max} vs. number of nodes.

fixed level due to saturation load.

From all these experiments, we can reach the conclusion that adaptive sub-channelization can improve the performance of IEEE 802.11 significantly. In our experiments we found up to 80% less collision in RTS messages, a throughput improvement of up to 40%, and decrease in delay up to 20%. We can hope that incorporation of AMC will increase the throughout of our new MAC further.

V. Conclusion

In this paper, we have demonstrated that dividing the nodes into discrete groups and allowing multiple concurrent transmissions can significantly reduce collisions in IEEE 802.11 DCF and improve the throughput of the WLAN. There are many application scenarios like data driven wireless networks, or wireless sensor networks with large number of nodes that can be directly benefitted from this research. In future, we will consider the impact of our proposed system on finite load conditions and adaptively adjusting the maximum number of channels based on system's load to achieve optimal throughput from the system. In that context, diving the nodes into equitable groups also remains as a major research challenge.

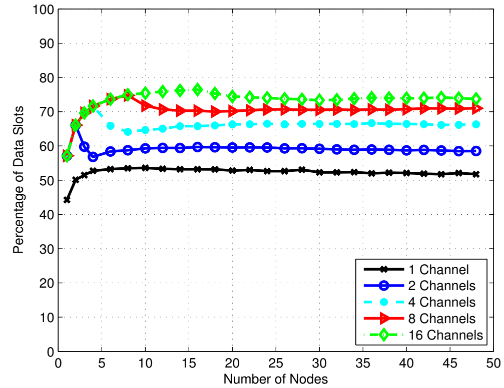

Fig. 10: Percentage of time the channel is used for carrying payload data vs. number of nodes for different c_{max}.

References

[1] M. S. et al., *Proposal for PAR and 5 criteria for very high throughput (VHT) SG for 60 ghz*, IEEE 802.11 Very High Throughput Study Group Std. doc. 08/223, May 2008.

[2] M. de Courville et al., *Proposal for PAR and 5 criteria for very high throughput (vht) sg for below 6 ghz operation*, IEEE 802.11 Very High Throughput Study Group Std. doc. 08/609, May 2008.

[3] J. Gross, M. Emmelmann, O. Punal, and A. Wolisz, *Dynamic multi-user ofdm for 802.11 systems*, IEEE 802.11 Very High Throughput Study Group Std. doc. 07/2062, July 2007.

[4] K. Sundaresan, R. Sivakumar, M. A. Ingram, and T.-Y. Chang, "A fair medium access control protocol for ad-hoc networks with mimo links," in *Proc. INFOCOM*, vol. 4, Mar. 7–11, 2004, pp. 2559–2570.

[5] R. R. Choudhury, X. Yang, R. Ramanathan, and N. H. Vaidya, "Using directional antennas for medium access control in ad hoc networks," in *MobiCom*, 2002, pp. 59–70.

[6] M. Morelli, C.-C. J. Kuo, and M.-O. Pun, "Synchronization techniques for orthogonal frequency division multiple access (ofdma): A tutorial review," *Proc. IEEE*, vol. 95, no. 7, pp. 1394–1427, July 2007.

[7] K.-D. Lee and V. C. M. Leung, "Fair allocation of subcarrier and power in an ofdma wireless mesh network," *IEEE J. Select. Areas Commun.*, vol. 24, no. 11, pp. 2051–2060, Nov. 2006.

[8] J. Gross, S. Valentin, H. Karl, and A. Wolisz, "A study on the impact of inband signaling and realistic channel knowledge for an example dynamic ofdm-fdma system," *Euro. Trans. Telecomms*, vol. 16, no. 1, pp. 37–49, 2005.

[9] S. Valentin, T. Freitag, and H. Karl, "Integrating multiuser dynamic ofdma into ieee 802.11 wlans - llc/mac extensions and system performance," in *Proc. IEEE ICC*, May 19–23, 2008, pp. 3328–3334.

[10] H. S. Lichte, S. Valentin, F. Eitzen, M. Stege, C. Unger, and H. Karl, "Integrating multiuser dynamic ofdma into ieee 802.11a and prototyping it on a real-time software-defined radio testbed," in *Proc. TridentCom*, May 21–23, 2007, pp. 1–9.

[11] M. Veyseh, J. J. Garcia-Luna-Aceves, and H. R. Sadjadpour, "Ofdma based multiparty medium access control in wireless ad hoc networks," in *Proc. IEEE International Conference on Communications ICC '09*, June 14–18, 2009, pp. 1–6.

[12] M. Veyseh and J. J. Garcia-Luna-Aceves, "Parallel interaction medium access for wireless ad hoc networks," in *Proc. 17th International Conference on Computer Communications and Networks ICCCN '08*, Aug. 3–7, 2008, pp. 1–6.

[13] 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.

[14] Y. Xiao and J. Rosdahl, "Throughput and delay limits of ieee 802.11," *IEEE Commun. Lett.*, vol. 6, no. 8, pp. 355–357, Aug. 2002.

[15] "Sim java 2, www.icsa.informatics.ed.ac.uk/research/groups/hase/simjava," 2009.

[16] B. OHara, *IEEE 802.11 handbook*, 2nd, Ed. IEEE Press.