Concurrent-MAC: Increasing Concurrent Transmissions in Dense Wireless LANs

Ghazale Hosseinabadi and Nitin Vaidya
Department of ECE and Coordinated Science Lab.
University of Illinois at Urbana-Champaign
{ghossei2, nhv}@illinois.edu

***Abstract*—This paper presents the design and performance evaluation of Concurrent-MAC, a MAC protocol for increasing concurrent transmissions and throughput in dense wireless LANs. Based on channel gain measurements and SINR values, sets of concurrent transmitters are identified by the backhaul of APs. A node gaining access to the channel, schedules one of its neighbors for concurrent transmission. Neighbor chosen for concurrent transmission can start transmitting on the channel, immediately after it overhears the privilege given to it for concurrent transmission. Our simulation results show that, in dense wireless LANs, Concurrent-MAC improves network throughput significantly compared to 802.11 DCF.**

I. INTRODUCTION

Deployment of Wireless Local Area Networks (WLANs) has grown rapidly in the past few years. IEEE 802.11 DCF is the MAC protocol commonly used in wireless LANs. 802.11 DCF employs *Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA)* scheme. In 802.11 DCF protocol, a node willing to transmit, senses the wireless channel to determine if the channel is busy or idle. If the channel is sensed busy, the node has to defer its transmission until the medium becomes idle. The main drawback of carrier sensing mechanism is that, the information regarding which node is transmitting on the channel is not considered. Some nodes might be eligible for concurrent transmissions while some might not, which is directly related to the SINR values at the receiver. Concurrent transmissions are transmissions that overlap in time. The ideal MAC protocol must 1) Prevent concurrent transmissions by interfering links. 2) Allow concurrent transmissions by non-interfering links. Maximizing the number of successful concurrent transmissions results in maximizing the aggregate throughput of wireless networks.

All CSMA/CA based MAC protocols suffer from the well-known "hidden terminal" and "exposed terminal" problems. The hidden terminal problem refers to terminals that do not sense each other's transmission, although their concurrent transmission results in collision at their corresponding receivers, due to excessive interference. The exposed terminal problem occurs where a node refrains from transmission on sensing the channel busy, even though its transmission can succeed concurrently with the ongoing transmission. As we will discuss in this paper, increasing the density of stations and APs in WLANs, increases the number of exposed terminals. A MAC protocol that identifies exposed terminals and enables concurrent transmissions by exposed terminals, improves network throughput in dense WLANs.

IEEE 802.11 protocol is designed based on a "single AP-multiple stations" architecture, in which each AP serves multiple stations and each station is associated with only one access point at a time. But in current deployments of WLANs, we observe that, in many cases, multiple APs are present in the vicinity of each station. [1] reports that in most hotspots, $3-20$ APs are present within the transmission range of each client. As wireless APs become cheaper, dense WLANs are more commonly found. The design of a MAC protocol that efficiently utilizes the presence of multiple APs to increase network throughput is an important problem in future deployments of WLANs.

In this paper, we design a MAC protocol, called *Concurrent-MAC*, which identifies and enables concurrent transmissions to improve network throughput in dense WLANs. Our protocol, Concurrent-MAC, exploits the presence of the infrastructure to find out which nodes can transmit concurrently. Each node is then given an accurate list of its concurrent neighbors. Whenever a node gains access to the channel, it gives a privilege to one of its neighboring exposed nodes to transmit concurrently. Priviledged neighbor can transmit concurrently, immediately after it overhears the privilege given to it for concurrent transmission. Concurrent transmissions are scheduled in a distributed manner, without the help of a centralized component. Only computing sets of concurrent transmitters is done in a central way.

The rest of this paper is organized as follows. We first review some related work in Section II. We present our protocol, Concurrent-MAC, in Section III. We compare the performance of Concurrent-MAC with IEEE 802.11 in Section IV.

II. RELATED WORK

The problem of coordinating multiple APs in WLANs has received significant attention over the past decade. Miu et al. [2] has designed the *Multi-Radio Diversity (MRD)* wireless network, which uses path diversity to improve throughput of

---

This research is supported in part by National Science Foundation award CNS 11-17539. Any opinions, findings, and conclusions or recommendations expressed here are those of the authors and do not necessarily reflect the views of the funding agencies or the U.S. government.

WLANs. Path diversity is based on the fact that the packet is transmitted over several different propagation paths and individual paths experience different levels of fading and interference. MRD has proposed a *frame combining* method which attempts to find the correct version of the transmitted frame even if it is received erroneously at all APs.

Zhu et al. [3] has proposed an AP association algorithm for deciding which APs to associate with. In uplink, their proposed heuristic selects an AP to be included in the set of associated APs of a station, if the addition of that AP will increase the throughput of the station by more than a threshold. The AP selection algorithm in downlink tries to balance the traffic load among the neighboring APs.

Different solutions have been proposed in the literature to solve the hidden and/or exposed terminal problem of 802.11 DCF protocol [4], [5], [6]. 802.11 protocol [4] defines a mechanism called RTS/CTS handshake to reduce collisions caused by the hidden nodes. [5] proposes a central scheduling algorithm, called CENTAUR, to manage downlink traffic in WLANs and to mitigate downlink hidden and exposed terminals problem. [6] proposes *CMAP (Conflict Maps)*, a MAC protocol for increasing the number of concurrent transmissions in wireless networks. In CMAP, initially, all stations transmit concurrently, even if their transmissions collide. Stations then measure the loss probability to figure out which nearby stations are interfering stations and which are exposed stations, based on which, the stations build conflicting transmissions map. A station willing to transmit on the channel, considers the current transmitters and consults the conflict maps, to decide whether to transmit or defer.

III. CONCURRENT-MAC DESIGN

Figure 1 shows the dense WLAN architecture we consider in this paper. Similar architecture has been proposed for dense WLANs in the literature [2], [3]. APs are connected to a central component called *controller* via a wired backbone. In Concurrent-MAC, the stations are not explicitly associated with the APs and a station's packets might be received by any of the nearby APs. Furthermore, any AP might transmit downlink packets to a station. The APs are configured such that they receive or overhear packets transmitted by close by stations. An AP forwards all the packets it receives or overhears to the controller, which filters redundant packets received by multiple APs and forwards only one copy to the higher network layers.

Our protocol, Concurrent-MAC has two major components: 1) A probe phase to determine the sets of concurrent transmitters. 2) An opportunistic token passing protocol to enable concurrent transmissions. In the following, we present detailed explanation of these two components.

A. Probe phase

The sets of concurrent nodes are determined by the controller during the probe phase. In this phase, all network nodes transmit a number of probe packets in a round robin manner. Every node overhears the probe packets transmitted by its nearby nodes. Comparing the transmit signal power of the packet transmitted by node i and the received signal power by node j, channel gain between i and j is calculated. We assume that the transmit power of a packet is included in the header of the packet. Channel gain information is collected by the set of stations and APs and is given to the central controller. Knowing the channel gains, the central controller computes the SINR for different combinations of nodes transmitting concurrently, to find out if their concurrent transmission can be received or captured successfully.

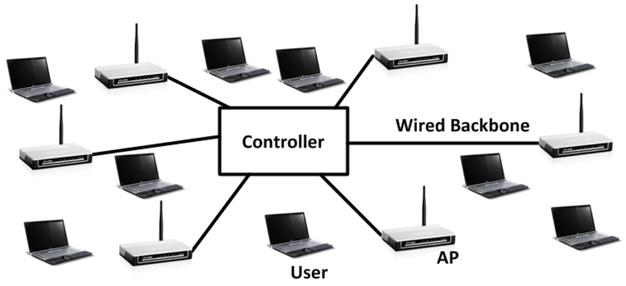


Fig. 1: System architecture

We note that in finding concurrent transmitters, concurrent transmission of acknowledgements is not considered. In Concurrent-MAC, acknowledgements to concurrent transmitters are not transmitted concurrently. The reason is that concurrent MAC ACKs might collide. Instead, the acknowledgements are transmitted sequentially. At the end of the probe phase, each node is given a list of its concurrent neighbors. Concurrent lists are computed periodically in order to prevent the stale information about the current channel conditions from degrading the protocol performance.

B. Enabling concurrent transmissions

Channel access mechanism of Concurrent-MAC is shown in Figure 2. As explained before, each node is given a list of its concurrent neighbors. In Concurrent-MAC, when node i transmits on the channel, it gives privilege to one of its neighbors, e.g. node j to transmit concurrently. Transmitting node i announces the privileged neighbor j in the privileged field of the MAC header of the data packets it transmits. By overhearing data packets, a privileged node is informed that it can transmit concurrently. If privileged node j was sensing the channel as idle before transmission of i starts and if it overhears the privilege given to it by node i, it will start transmitting on the channel immediately. Non-privileged nodes (e.g. node k in Figure 2) have to defer their transmission till when the channel becomes idle. This process of giving a privilege to a neighboring node repeats in each transmission. Whenever a node transmits on the channel, a concurrent transmission might start. On the other hand, in IEEE 802.11 DCF protocol, when a node starts transmission, its nearby nodes do not transmit concurrently, since they sense the medium as busy, although their concurrent transmission might be received or captured successfully.

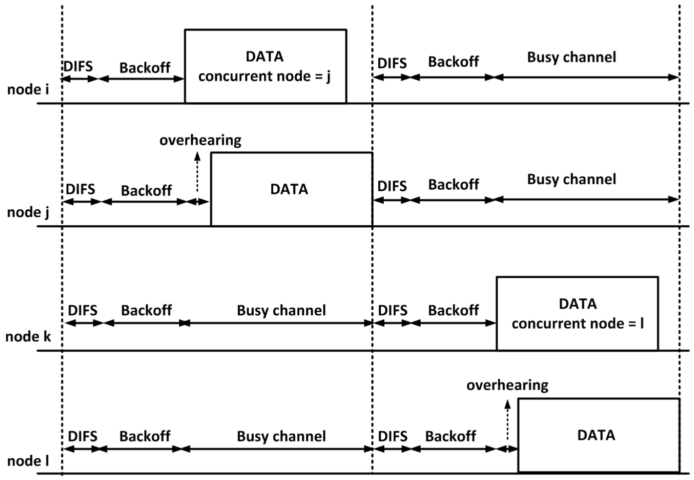


Fig. 2: Access method of Concurrent-MAC protocol

If opportunistic overhearing does not work, i.e., privilege is not received by the privileged node, Concurrent-MAC operates similar to 802.11 DCF. But when the privileged node overhears the token, it can transmit concurrently. Signaling mechanism in Concurrent-MAC is done via embedding the information regarding queue length and privileged neighbor in the header of data packets by the source node and overhearing the packets to retrieve such information by the neighboring nodes.

In Concurrent-MAC, each node keeps track of queue length of its neighboring nodes. A node that gains access to the channel assigns a privilege to one of its concurrent neighbors with non-zero queue length (i.e., with backlogged traffic). If privilege is assigned to a node with an empty queue, the privileged node would not take its chance to transmit concurrently. This results in under-utilization of the wireless channel. As long as the privilege is assigned to a node with backlogged traffic, the privileged node can immediately transmit a packet concurrently with the current transmitter.

Since the wireless channel is a shared medium, node j might overhear packets that are not intended for it, i.e., packets with destination address different from j. If the overhearing node is chosen as the priviledged node, it cancels its backoff counter, if its backoff counter is running or paused, and immediately transmits on the channel, if it has any backlogged traffic. Since nodes transmit packets with different sizes, transmission time of one packet might be different for different nodes. From the time that the priviledged node j received the priviledge from node i, it can transmit data packets for a duration of `txtime`, where `txtime` is the duration of one packet transmission by node i. During this time, priviledged node is allowed to transmit as many packets as it can. If one packet transmission by node j lasts longer than `txtime`, node j is not allowed to transmit concurrently with node i.

The scheduling algorithm of Concurrent-MAC provides network nodes with a mechanism for choosing a concurrent neighbor from all possible concurrent neighbors computed by the controller. Different scheduling algorithms can be used for choosing the privileged neighbor. Here, we present two example scheduling policies. In the first policy, a transmitting node chooses one of its concurrent neighbors uniformly at random. Another policy is to assign appropriate probabilities to different concurrent neighbors such that each network node has the same probability of being chosen for concurrent transmission.

A node can give privilege to one of its neighbors, only if it has gained access to the channel via the backoff mechanism and not via obtaining privilege from another node. A privileged node is not allowed to assign privilege to another node, since, due to excessive interference, the transmission by the union of the privileged nodes might not result in successful reception.

Our protocol, Concurrent-MAC, enables concurrent transmissions in both uplink and downlink. Concurrent-MAC, enables stations to transmit concurrently, if their concurrent transmissions can be received or captured successfully by the backhaul of APs. Similarly, two APs can transmit concurrently, if their concurrent transmissions can be received or captured successfully by two different stations. A station and an AP can transmit concurrently, if their concurrent transmissions can be successfully received or captured by an AP and a station, respectively.

TABLE I: Parameters used in the simulation study

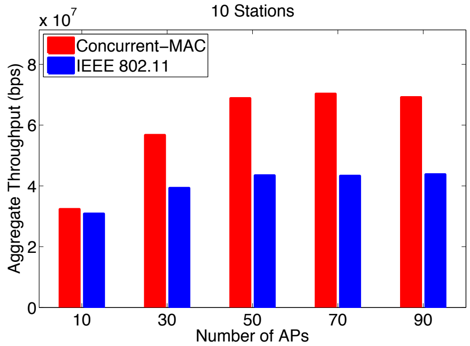
Propagation	Shadowing
RTS/CTS	disabled
Transmit power	15 dBm
Thermal noise	-93 dBm
Carrier sense threshold	-91dBm
SIFS	16μsec
DIFS	34 μsec
Slot time	9μsec
CWmin	15
CWmax	1023
Data rate	54 Mbps

IV. EVALUATION

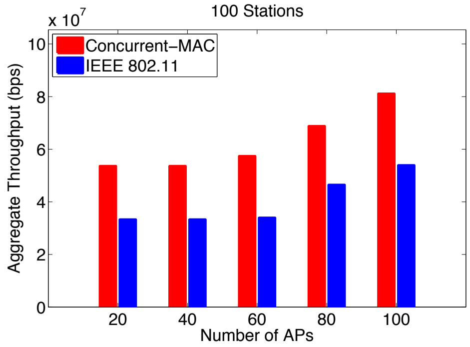
We simulate Concurrent-MAC and 802.11a in ns-2 to compare performance of these two MAC protocols. Table I reports the configuration parameter values of the wireless network analyzed in this section. The network is a wireless LAN in which stations and APs are placed uniformly at random in a square area. IEEE 802.11 RTS/CTS mechanism is turned off. We use a log-distance path loss model with path loss exponent of 4.02 to simulate the indoor office environment [7]. Packet payload size is 1500 bytes. Each simulation lasts for 30 seconds and the presented results are averaged over 5 runs.

A. Networks with fully backlogged CBR traffic

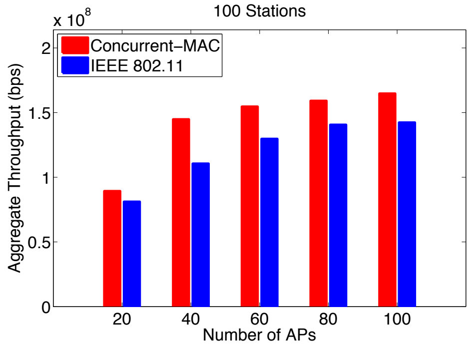
We first simulate Concurrent-MAC and IEEE 802.11a under fully backlogged CBR traffic, in which there is always backlogged packets in the transmission queue of each transmitter.



(a) area=17mx17m



(b) area=50mx50m



(c) area=100mx100m

Fig. 3: Fully backlogged CBR traffic

We place 10 stations, uniformly at random, in a square area of $17m$x$17m$. Under our simulation parameters, the carrier sense range is 25m. With this area size, the deployed network is a single contention domain in which every station senses every other transmission and only one transmission is possible in IEEE 802.11 protocol, at each time instance. We vary the number of APs from 10 to 90 to investigate the effect of increasing number of APs.

For this network, aggregate throughput of Concurrent-MAC and 802.11 is shown in Figure 3(a). As we see in this figure, the throughput improvement obtained by Concurrent-MAC is between 44% and 62% with 30-90 APs. We note that in our simulations, APs are placed randomly in the area, and not all of them are required for successful reception of concurrent transmissions. For example, we found out that for this network topology and for the case of 90 APs present in the area, only 11 APs are needed to successfully receive all concurrent transmissions and to maximize network throughput in Concurrent-MAC. We believe that determining the placement of APs in order to maximize concurrent transmissions and network throughput is an interesting research problem.

Figures 3(b) and 3(c) plot the throughput for two larger networks, with multiple contention domains. In these figures, 100 stations and 20-100 APs are placed uniformly at random in a square area. Network size is $50m$x$50m$ in Figure 3(b) and $100m$x$100m$ in Figure 3(c). These figures show that throughput improvement obtained by Concurrent-MAC, compared to 802.11, is between 10% and 69%.

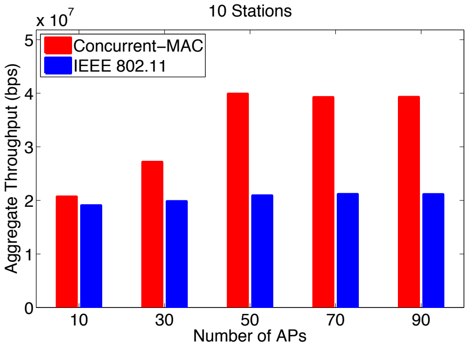
B. Networks with TCP Traffic

In this section, we identify performance of Concurrent-MAC when network traffic is TCP. When stations traffic is TCP, two types of application packets are transmitted in the network: 1) TCP packets transmitted from stations to APs, and 2) TCP acknowledgements transmitted from APs to stations. We perform simulations for two network sizes, i.e., 17mx17m and 50mx50m. APs and stations are placed uniformly at random in the area. In this set of simulations, any node might transmit concurrently with any other node, where a node might be a station transmitting TCP packets or an AP transmitting TCP acknowledgements. Figure 4 shows the aggregate throughput of 802.11a and Concurrent-MAC for these two area sizes. When traffic is TCP, although the buffer of stations and APs might not be fully backlogged, network nodes might have few packets backlogged in their transmission queue, in which case privilege can be given to neighboring nodes for concurrent transmissions. This results in increased network throughput achieved by Concurrent-MAC, compared to 802.11 protocol.

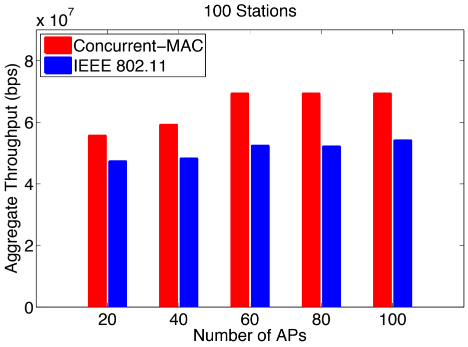
C. Decreasing Transmission Rate to Increase Concurrent Transmissions

As we have found in this work, decreasing transmission rate of nodes might result in increasing the number of concurrent transmissions in the network. The reason is that by decreasing transmission rate, the SINR threshold for reliable data reception decreases. In this case, nearby nodes might be able to transmit concurrently, at a lower rate, if their concurrent transmissions are received with SINR higher than the SINR threshold corresponding to the lower rate. This means that some sets of nodes that can not transmit concurrently at their highest transmission rate, might be able to transmit concurrently at lower rates. In a single contention domain, if total throughput of nodes transmitting concurrently, at lower rates, is more than the throughput of nodes transmitting individually, with their maximum possible rate, concurrent transmission at lower rates increases aggregate throughput of Concurrent-MAC, compared to 802.11.

In figure 5, we consider a single contention domain in which three stations and three APs are placed such that concurrent



(a) area=17mx17m



(b) area=50mx50m

Fig. 4: TCP traffic

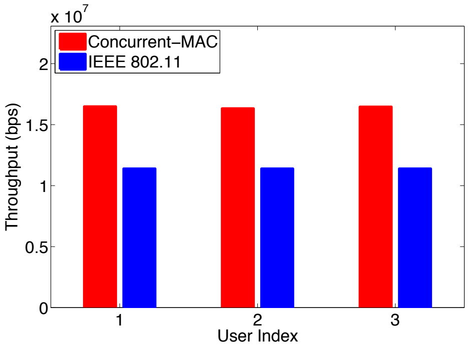


Fig. 5: Decreasing transmission rate to increase concurrency

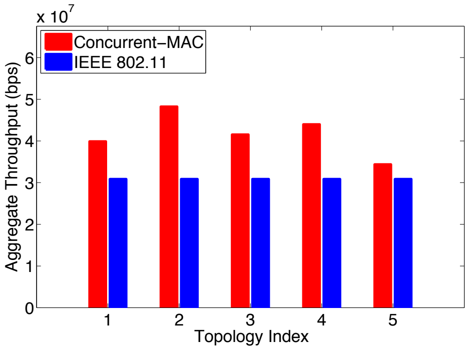


Fig. 6: Aggregate Throughput (5 topologies of size 6mx6m)

transmission is not possible at highest transmission rate of stations, i.e., 54 Mbps. But if stations decrease their transmission rate to 24 Mbps, they are able to transmit concurrently. As shown in this figure, throughput of all stations are improved by Concurrent-MAC, when stations decrease their transmission rate from 54 Mbps to 24 Mbps.

We then consider 5 different topologies in which 5 stations and 5 APs are placed uniformly at random in an area of size $6m$x$6m$. This area size is close to the size of an office or a conference room. In these 5 topologies and in Concurrent-MAC protocol, concurrent transmission at rate 54 Mbps is not possible. We note that since concurrent transmission at rate 54 Mbps is not possible in these topologies, Concurrent-MAC and 802.11 perform the same, if nodes only transmit at rate 54 Mbps. Figure 6 shows the aggregate throughput of Concurrent-MAC and 802.11 protocol, where in Concurrent-MAC, stations transmit concurrently at lower rates, if transmitting concurrently at lower rates results in higher total throughput, compared to transmitting individually at rate 54 Mbps. The transmission rate options we consider in this simulation, are the 8 options possible in 802.11a protocol. Figure 6 shows an improvement of 11%-56%, which is achieved by decreasing transmission rate to increase concurrency.

V. CONCLUSION

In this paper, we presented the design and performance evaluation of *Concurrent-MAC*. Concurrent-MAC is a MAC protocol that uses an opportunistic overhearing mechanism to schedule network nodes for concurrent transmissions in dense WLANs. The main design goal of Concurrent-MAC is to increase aggregate throughput by identifying and enabling successful concurrent transmissions. Our ns-2 simulations show that Concurrent-MAC can achieve significant improvement in system throughput compared to 802.11 DCF.

REFERENCES

[1] A. Akella, G. Judd, S. Seshan, and P. Steenkiste, “Self-management in chaotic wireless deployments,” In *MobiCom ’05*, August 2005.

[2] A. Miu, H. Balakrishnan, and C. E. Koksal, “Improving loss resilience with multi-radio diversity in wireless networks,” In *Proc. of ACM MobiCom*, pp. 16-30, August 2005.

[3] Y. Zhu, Z. Niu, Q. Zhang, B. Tan, Z. Zhou, and J. Zhu, “A multi-AP architecture for high-density WLANs: protocol design and experimental evaluation,” in *Proc. of IEEE SECON,* pp. 28-36, June 2008.

[4] IEEE 802.11, *Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications,* IEEE standard for information technology telecommunications and information exchange between systems-local and metropolitan area networks-specific requirements, 2007.

[5] V. Shrivastava, N. Ahmed, S. Rayanchu, S. Banerjee, S. Keshav, K. Papagiannaki, and A. Mishra, “CENTAUR: Realizing the Full Potential of Centralized WLANs through a Hybrid Data Path.” In Proc. of *the 15th annual international conference on Mobile computing and networking,* pp. 297-308, 2009.

[6] M. Vutukuru, K. Jamieson, and H. Balakrishnan, “Harnessing exposed terminals in wireless networks,” in *USENIX NSDI*, June 2008.

[7] D. B. Faria, “Modeling signal attenuation in IEEE 802.11 wireless lans-Vol. 1,” Stanford University, 2005.