

Comparative Investigation on CSMA/CA-Based Opportunistic Random Access for Internet of Things

Chong Tang, *Student Member, IEEE*, Lixing Song, Jagadeesh Balasubramani, Shaoen Wu, *Member, IEEE* Saad Biaz, *Member, IEEE* Qing Yang, *Member, IEEE* and Honggang Wang, *Senior Member, IEEE*

Abstract—Wireless communication is indispensable to Internet of Things (IoT). CSMA/CA is a well-proven wireless random access protocol and allows each node of equal probability in accessing wireless channel, which incurs equal throughput in long term regardless of the channel conditions. To exploit node diversity that refers to the difference of channel condition among nodes, this paper proposes two *opportunistic* random access mechanisms: *overlapped contention* and *segmented contention*, to favor the node of the best channel condition. In the overlapped contention, the contention windows of all nodes share the same ground of zero, but have different upper bounds upon channel condition. In the segmented contention, the contention window upper bound of a better channel condition is smaller than the lower bound of a worse channel condition; namely, their contention windows are *segmented* without any overlapping. These algorithms are also polished to provide temporal fairness and avoid starving the nodes of poor channel conditions. The proposed mechanisms are analyzed, implemented and evaluated on a Linux-based testbed and in the NS3 simulator. Extensive comparative experiments show that both opportunistic solutions can significantly improve the network performance in throughput, delay, and jitter over the current CSMA/CA protocol. In particular, the overlapped contention scheme can offer 73.3% and 37.5% throughput improvements in the infrastructure-based and ad-hoc networks, respectively.

Index Terms—CSMA/CA, opportunistic access.

I. INTRODUCTION

Internet of Things (IoT) is an ambitious plan, also the ultimate goal and result of the integration between information technologies and physical objects/system in the world. To interconnect everything in IoT, wireless communication plays a critical role in many scenarios such as user access, sensor networking, mobility, etc. Carrier sensing multiple access/collision avoidance (CSMA/CA) is widely used in many random access wireless networks as it can reduce

Chong Tang is with the Department of Computer Science, University of Virginia, Charlottesville, VA, 22903 USA. E-mail: ctang@virginia.edu. This work was done when he was with University of Southern Mississippi

Lixing Song and Shaoen Wu are with the Department of Computer Science, Ball State University, Muncie, IN, 47306 USA. E-mails: {slong, swu}@bsu.edu.

Jagadeesh Balasubramani and Saad Biaz are with Department of Computer Science and Software Engineering, Auburn University, Auburn, AL, 36849 USA. E-mail: zb0047@tigermail.auburn.edu, biazsaa@auburn.edu.

Qing Yang is with Computer Science Department, Montana State University, Bozeman, MT, 59715 USA. E-mail: qing.yang@cs.montana.edu.

Honggang Wang is with Department of Electrical and Computer Engineering, University of Massachusetts Dartmouth, Dartmouth, MA, 02747 USA. E-mail: hwang1@umassd.edu.

Copyright (c) 2012 IEEE. Personal use of this material is permitted. However, permission to use this material for any other purposes must be obtained from the IEEE by sending a request to pubs-permissions@ieee.org.

potential collisions and, thereby, improve the overall network performance. It allows each node of equal probability in accessing wireless channel, which incurs equal throughput in long term regardless of the channel conditions. However, in wireless networks, nodes' channels oscillate between good and poor conditions. How to take advantage of the node diversity in random access wireless networks for high performance and fairness is interesting, challenge but beneficial, which is referred as opportunistic communication in traditional coordinated wireless networking.

The signal transmitted on a wireless channel may reach the destination directly (line of sight) or through multiple reflections on local scatterers (buildings, mountains, etc.). As a result, wireless channel conditions are affected by various random attenuations and delays. Moreover, the mobility of nodes or scattering environment may cause these random fluctuations to vary with time. Therefore, wireless nodes often experience different channel conditions, which is referred as *spatial diversity* or *node diversity*. Due to the spatial diversity, a wireless node with excellent channel condition may be able to transmit data at the highest bit rate while others with poor links may not be able to transmit any data even at the lowest rate. Channel condition variations also lead to *time diversity*: a node may have a link of high bit rate at a moment, but may have to use a low rate thereafter. The node diversity of channel conditions is often considered detrimental in traditional wireless communication systems, especially in random access networks, because the channel condition is unpredictable and uncontrollable.

In recent years, *opportunistic* approaches have been attempted to exploit the inherent randomness of wireless channel to improve random access wireless network performance and utilization, including opportunistic rate adaptation [10], transmission [1], scheduling [11], [12], [13] and routing [14], [15], [16], [17]. Opportunistic protocols exploit node diversity by granting higher priority to nodes with good channel conditions and/or time diversity by extending the use of channel in good conditions.

Unlike the above-mentioned opportunistic *transmission* schemes, we propose and evaluate two opportunistic *access* schemes that exploit *node diversity* in the CSMA/CA. These schemes enable the node with the best channel condition to have the greatest probability of accessing the shared channel, but do not starve the nodes with poor links. In the long run, each node probably obtains a throughput proportional to its channel conditions. The main contributions of this work consist of: 1) two distributed opportunistic random access

variants, 2) theoretical analysis of their opportunism, 3) the development of a Linux-based testbed with the proposed algorithms implemented, and 4) extensive performance evaluation on testbeds and in *NS3* simulator.

The rest of this paper is organized as follows. CSMA/CA and related opportunistic schemes are briefly reviewed in Section II. Then, Section III presents the problems that motivate this work. Next, Section IV discusses the proposed opportunistic access variants and analyzes the opportunism of the proposed schemes. Section V presents the implementation of a testbed, experiment settings, and the performance evaluations on the testbed and the *NS3* simulator. Finally, Section VI concludes this paper.

II. RELATED WORK

A. CSMA/CA

Many wireless networks such as IEEE 802.11 [6] adopt CSMA/CA [4] as the core media access mechanism. The opportunistic algorithms proposed in this work are therefore based on this mechanism. CSMA/CA is a contention-based MAC protocol. In a typical CSMA/CA network, regardless of ad-hoc or infrastructure mode, all nodes that have data to transmit on the shared wireless link must undergo a contention procedure first. Only the node that wins the contention can transmit while all others freeze the contention procedure until the winner completes its transmission. The contention is regulated by a *binary exponential backoff* process in which each node maintains a contention window that has a lower bound of “0” and an upper bound that starts with an initial value and may exponentially increase when network becomes congested. Then, a node that is ready to transmit randomly selects a backoff value T_b from the contention window using a *uniform* distribution. The node will keep sensing the channel. If the channel is busy, the backoff value is frozen until the channel becomes idle. Otherwise, it is decremented by one at every (idle) time slot. When the backoff value reaches “0”, the node starts its transmission. If the transmission fails, the current contention window upper-bound is doubled unless it has reached the maximum. Then another backoff procedure is repeated with the updated contention window.

B. Opportunistic Scheduling

An opportunistic CSMA/CA was proposed by Hwang and Cioffi [23], which targets the node diversity in WLAN. Their algorithm schedules a node to transmit at a specific time slot according to the signal-to-noise-ratio (SNR) of its channel. To avoid the nodes with the same SNR to collide with each other on the same time slot, a random number is introduced when the time slot is determined. Zhai *et al.* [20], [10] proposed an opportunistic media access control protocol that focused on the opportunistic scheduling of traffic at a node to its neighbors based on their channel conditions. The traffic to the neighbor with the best channel conditions is scheduled first for transmission.

C. Opportunistic Transmission

Opportunistic transmission based on multiple rates was demonstrated to yield a significant improvement of the network performance in IEEE 802.11 networks [1], [2], [3] where a node opportunistically transmits multiple frames if its bit rate is high, instead of traditionally one frame. These algorithms rely on rate adaptations such as RBAR [5] to estimate the channel conditions in terms of bit rate and then a sender calculates the number of frames that should be transmitted based on the adapted bit rate to maintain temporal fairness among nodes. Opportunistic transmission occurs after the contention of channel access that is still conducted with traditional non-opportunistic approach: equal probability access. It exploits the time diversity of a node, not the node diversity of a network. Vaidya [21], [22] proposed to dynamically vary the contention window to achieve the distributed proportional scheduling in IEEE 802.11 WLANs. Based on the weight assigned to each traffic flow, the contention window is calculated such that a more weighted flow has a smaller contention window. Then, the flow has more chances to use the channel for delivering more data than less weighted flows.

III. PROBLEM STATEMENT AND MOTIVATION

The motivation of this work is to exploit *node diversity* in random access wireless networks. This motivation stems from a few observations as described below.

A. Opportunistic Access in Wireless Networks

The access mechanism in a wireless network with node diversity has significant impact on the network performance and channel utilization. Let us examine a network with a base station and two client nodes *A* and *B*. Suppose *A* has poor channel conditions supporting a bit rate of R_A and *B* has good channel conditions supporting a bit rate of R_B . In one extreme case where only *A* has the access to the channel, the network throughput is R_A assuming no packet failure. In the other extreme case where only *B* uses the channel, the network throughput is R_B . Otherwise, if *A* and *B* share the channel, the network throughput will be some value between R_A and R_B . Namely, R_B and R_A are respectively the upper and lower bounds of the network throughput. Therefore, to improve the network throughput and channel utilization, *B* should be favored for accessing the channel, which is referred as *opportunistic access*.

B. Equal-Probability Access in CSMA/CA

In CSMA/CA, regardless of the channel conditions, all nodes have *equal* probabilities of accessing the channel because they *uniformly* select the backoff value from the *same-size* contention windows. In a two-node network with CSMA/CA, a node *A* with poor channel conditions may sometimes beat the node *B* with good channel conditions because both of them have equal probabilities of access the channel. From the discussion in the above section, however, it is more beneficial to allow *B* to transmit for two reasons. First, *B* would use the wireless channel more efficiently because

it takes less time in transmitting the same amount of data than A does. This opportunism can lead to higher utilization, efficiency, and throughput of the overall network. Second, due to the inherent temporal variations of wireless channels, B cannot occupy the channel forever, i.e. A will win the channel access when its channel conditions become better.

C. Opportunistic Transmission vs. Opportunistic Access

Each communication cycle in the random access wireless networks can actually be considered as two phases: *access* (contention) followed by *transmission*. Opportunistic transmission focuses on the transmission phase but not the access phase. Although opportunistic transmission improves the network performance by exploiting time diversity, it does not guarantee the node with the best channel condition has the best chance to win the channel. However, the node with the best channel conditions deserves the chance to use the channel because its channel may degrade later. To improve the utilization of scarce wireless resources, we are motivated to design opportunistic *access* algorithms that grant the channel in probability to the node that has the best channel condition, namely that is most likely to generate the largest instantaneous network throughput.

With the observations discussed above, to improve network efficiency and channel utilization, we are motivated to design distributed opportunistic access algorithms that probabilistically favor the nodes with best channel conditions in winning the channel contention.

IV. DESIGN OF OPPORTUNISTIC RANDOM ACCESS

To achieve opportunistic random access, we propose two variants based on the contention mechanism in the CSMA/CA. Both of them target at calculating the contention window for each node based on its instantaneous channel conditions. In addition, opportunistic access inherently tends to starve nodes with poor channel conditions. These schemes are also enhanced to address the starvation problem with temporal fairness.

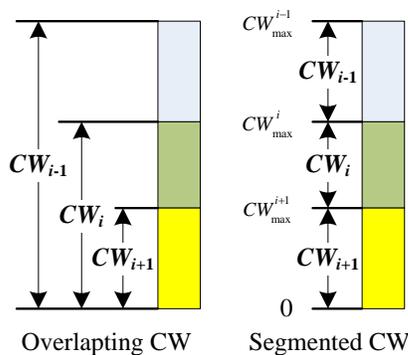


Fig. 1: Illustration of Contention Windows

A. Overlapped Contention

The following presents two variants of opportunistic access based on the contention window mechanism in current IEEE 802.11 standard. In the *overlapped contention* approach, the contention windows of all nodes share the same lower bound

$CW_{min} = 0$ as CSMA/CA does, but have different initial upper bounds CW_{max} that are determined by channel conditions measured by achievable bit rates. Therefore, the contention windows of all nodes overlap in ranges as plotted on the left part of Fig. 1. The initial CW_{max} is inversely proportional to the ratio of current achievable bit rate over the basic bit rate, which is computed as:

$$CW_{max} = \lceil \alpha \times \frac{R_b}{R_i} \times CW_b \rceil \quad (1)$$

where R_i refers to the current achievable bit rate of a particular node i , R_b denotes the basic rate in a bit rate set and CW_b is the default CW value defined in CSMA/CA, e.g. 15 in IEEE 802.11n. Note that R_b/R_i may be very small, for example $6.5/600 = 0.01$ in IEEE 802.11n. Therefore, a scaling factor α is introduced to make sure that the CW_{max} for the highest bit rate is no less than a certain value. From this formula, a high bit rate (with good channel conditions) leads to a small CW_{max} and thereby a higher chance to win the channel contention. Then, the computed CW_{max} can be used by the *binary exponential backoff* procedure in the CSMA/CA to fulfill the opportunistic access.

Note that, in the *overlapped contention*, a node with low bit rate still has the probability to beat another node with a high bit rate, i.e. the lower rate node still has chance to select a smaller backoff value because their contention windows have the same lower bound of “0”.

1) *Opportunism of Overlapped Contention*: As discussed above, the *overlapped contention* is an opportunistic access that assigns larger probability of winning the channel to the node with higher rate. Denote P_e^i as the probability that a node i attempts to transmit a frame. According to [7], with different contention window size, the P_e^i is calculated as:

$$P_e^i = \frac{2}{CW_{max}^i + 1} \quad (2)$$

where CW_{max}^i is the contention window size of node i . Define P_t^i the successful transmission probability of node i . Then

$$P_t^i = P_e^i \cdot \prod_{j \neq i} (1 - P_e^j) \quad (3)$$

Define $\frac{P_t^i}{P_t^j}$ as the opportunism metric that node i will compete node j in accessing the channel. Based on Eq. (1), Eq. (2) and Eq. (3), we have

$$\begin{aligned} \frac{P_t^i}{P_t^j} &= \frac{\Delta(\lambda - R_j)}{\lambda - \Delta R_j} \\ &= 1 + \frac{(\Delta - 1)\lambda}{\lambda - R_i} \end{aligned}$$

where $\Delta = \frac{R_i}{R_j}$ and $\lambda = \alpha \cdot CW_b \cdot R_b$. According to Eq. (1),

$$\lambda = CW_{max}^j \times R_j \quad (4)$$

Because the contention window CW_{max}^j is always larger than 1, we have $\lambda > R_j$. Assume that node i has the largest bit rate R_i in the networks, then we have $\Delta > 1$ and also $\frac{P_t^i}{P_t^j} > 1$. This clearly shows that the *overlapped contention opportunistic access helps those nodes at higher rates to win the contention*.

B. Segmented Contention

To strictly grant a higher priority of accessing channel to the nodes with better channel conditions, another algorithm *separates* the contention windows for nodes at different bit rates as illustrated on the right of Fig. 1. In this approach, the initial contention window is still computed as in Eq. (1). However, unlike the *overlapped contention* that maintains the same $CW_{min} = 0$ for all contention windows, this algorithm differentiates the CW_{min} for different channel conditions. A higher bit rate results in an CW_{max} smaller than the CW_{min} of a node at a lower bit rate.

According to Eq. (1), the CW_{max} of every node can be computed and sorted, as shown in the right of Fig. 1. In this figure, CW_{max}^i and CW_{max}^{i+1} denote the computed initial upper bounds of CWs for bit rate R_i and R_{i+1} , respectively. Assume $R_i < R_{i+1}$, we can let $CW_{min}^i = CW_{max}^{i+1}$, so the contention windows of all nodes will form a segmentation in the entire contention window. The segmentation of contention windows can guarantee that nodes with higher channel conditions R_{i+1} can never get time slots smaller than a node at lower bit rate R_i .

The *segmented contention* approach can be considered semi-probabilistic since 1) the access of nodes with the same bit rate is random as they have the same initial window size, but 2) the access of nodes with different rates is deterministically prioritized as lower-rate nodes never get smaller backoff values than higher-rate nodes. This approach provides a tight opportunism by grouping nodes with similar channel conditions into the same random access team. Note that this approach leads to a significant problem: starvation of nodes with poor conditions which will be addressed in Section IV-C.

1) *Comparison between Segmented and Overlapped Opportunistic Accesses:* The *segmented contention* deterministically guarantee that nodes at higher rates will win the lower rate nodes because the latter can never have chance to obtain a smaller backoff window than the former. Therefore, the opportunism is deterministically maintained. In this section, we compare the system performance of *segmented* and *overlapped* contentions while both provide opportunism in access.

For comparison, we define a system performance metric $\chi = \frac{R}{E(T_b)}$, where R refers to the bit rate of a node that wins the access and $E(T_b)$ is the expected backoff time slots before transmission. Clearly, larger the value of χ , the better the network performance. χ serves as an network efficiency metric to evaluate the system performance of contention based opportunistic variants.

In the *segmented contention*, nodes at different bit rates do not share the same lower or higher bounds. Refer to the right plot of Fig. 1. The expectation of backoff for node i becomes $E(T_b) = (CW_{min}^i + CW_{max}^i)/2$, where CW_{min}^i is the lower bound of node i 's contention window, which is also the higher bound of node $i+1$. So if node i runs the *segmented contention* algorithm, we have its efficiency metric as:

$$\chi_s^i = \frac{2R_i}{CW_{max}^i + CW_{max}^{i+1}} = \frac{2\theta(R_i)^2}{\lambda} \quad (5)$$

where $\theta = \frac{\epsilon}{1+\epsilon}$ and $\epsilon = \frac{R_i}{R_{i+1}}$.

In the *overlapped contention*, a node at higher bit rate does not deterministically beat a lower bit rate node because of the overlapped contention windows. Consider two nodes i and j having bit rates ($R_i > R_j$). Denote P_i the probability that node i wins over j in channel access. Similarly, P_j refers to the probability that node j wins over i in channel access. Then, P_j and P_i can be calculated as follows:

$$P_j = \sum_{x=1}^{CW_i} \frac{1}{CW_j} \cdot \frac{CW_i - x}{CW_i} \approx \frac{CW_i}{2CW_j}$$

$$P_i = \frac{CW_j - CW_i}{CW_j} + \frac{CW_i}{CW_j} \sum_{x=1}^{CW_i} \frac{1}{CW_i} \cdot \frac{x-1}{CW_i}$$

$$\approx 1 - P_j$$

where $CW_j = CW_{max}^j$ and $CW_i = CW_{max}^i$. Therefore, the efficiency metric of node i with *overlapped contention* scheme can be derived as:

$$\chi_o^i = P_i \cdot \frac{2R_i}{CW_i} + P_j \cdot \frac{2R_j}{CW_j} \quad (6)$$

By substituting Eq. (1) into Eq. (5) and (6), we can obtain:

$$\frac{\chi_o^i}{\chi_s^i} = \frac{\Delta^2 + 2(\Delta - 1/2)}{2\theta\Delta^3} \quad (7)$$

In this equation, ϵ takes values from $[1.1, 2]$ in 802.11n, then θ will be constants falling into $[0.5, 0.7]$. Therefore, Eq. (7) is a monotonic decreasing function of Δ . It decreases to 1 when $\Delta \approx 1.5$ and approaches to 0 hereafter. This means that the *segmented contention* outperforms *overlapped contention* when $\Delta > 1.5$ that happens at the probability of 85% in IEEE 802.11 with rates (13, 26, 39, 52, 78, 104, 117, and 130 Mb/s) for 20 MHz bandwidth if two nodes have different bit rates.

C. Temporal Fairness to Avoid Starvation

The equal probability access regardless of channel conditions in the CSMA/CA leads to an anomaly that all nodes will have identical throughput in the long run [7], which is called *throughput fairness*. It is obvious that the nodes at lower bit rate hurt the throughput of the nodes at higher bit rates as well as the overall throughput of the network. This fairness is not "fair" in temporal use of the channel among nodes. With the identical throughput, a node at the lowest bit rate in a network will use the channel for the longest time. On the other hand, as discussed in Section III, although opportunistic access can improve the network throughput by always favoring the nodes with the best channel conditions, it may starve the nodes with poor channel conditions. A solution to both problems of identical throughput and starvation is to achieve *temporal fairness* among nodes [18] that is defined as each node has approximately the identical amount of time in using channel.

To achieve temporal fairness, we propose to use a *bit rate normalized average throughput* as a metric in computing the initial contention window. Each node keeps tracking the average throughput in an exponentially weighted interval t_w . Suppose node s is the transmitter and node i has packets to transmit, then the average throughput of i is updated in each time slot with a low-pass filter as:

$$T_i[m+1] = \begin{cases} \left(1 - \frac{1}{t_w}\right) \times T_i[m] + \frac{1}{t_w} \times R_i[m+1], & i = s \\ \left(1 - \frac{1}{t_w}\right) \times T_i[m], & i \neq s \end{cases} \quad (8)$$

The *bit rate normalized average throughput* for node i having bit rate R_i in the m -th interval is defined as:

$$T_i^*[m] = T_i[m]/R_i \quad (9)$$

This value will be used to compute the initial contention window. As a result, Eq. (1) is accordingly updated as:

$$CW = \alpha \times T_i^*[m] \times CW_b \quad (10)$$

The contention with Eq. (10) maintains two important features: temporal fairness and opportunism. The temporal fairness is achieved because the bit rate normalized average throughput can actually be explained as the *temporal quota* of a node in the transmission period. This is clear if we rewrite the definition of $T_i^*[m]$ as:

$$T_i^*[m] = (t_c \times T_i[m]/R_i)/t_c \quad (11)$$

where t_c is a period of length. Therefore, $t_c \times T_i[m]$ is the average transmitted data in bits and, $t_c \times T_i[m]/R_i$ is the transmission time. Thus, $(t_c \times T_i[m]/R_i)/t_c$ is the percentage of the time that node i being able to transmit in the time period t_c .

The opportunism in Eq. (9) is driven by the bit rate. If a node has a high bit rate, its $T_i^*[m]$ tends to be small. According to Eq. (10), a small $T_i^*[m]$ leads to a small contention window CW . If a node often wins the contention, it will end up with a very large average throughput $T_i[m]$. Due to a large $T_i^*[m]$, its contention window will be enlarged and its chance to win the channel decreases.

Note that the size of the weighted interval, t_w , is associated with the latency requirement of applications. If t_w is large, it allows the node with the optimal channel condition to use the channel for a long duration, but may hurt other nodes having applications requiring low latency. If it is small, the channel is frequently switched among nodes of different bit rates and the overall performance degrades. Another concern is the support of QoS. If multiple classes of applications are involved, each class has different requirements, especially regarding latency. Then, the basic contention window size CW_b for each packet will be updated according to its priority.

V. PERFORMANCE EVALUATION

A. Evaluation with Prototyping

To evaluate the opportunistic access performance in real world, we implemented a Linux based testbed. The implementation includes only the first algorithm, *overlapped contention*, because the other two require the modification of the lower bound of the backoff window that is “sealed” as 0 in the closed firmware to which we do not have access.

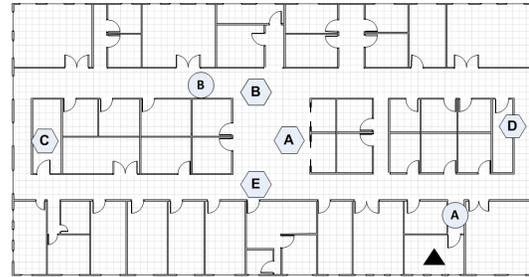


Fig. 2: Floor Plan

1) *Platform and Experimental Settings*: Our prototype testbed consists of six embedded computing nodes based on Alix 3dc SBC that has a 512 MHz AMD LX800 CPU, 256M RAM, and 8GB SD Card storage, and runs Linux kernel version 3.3.1. Each node is equipped with a miniPCI IEEE 802.11a/b/g/n WiFi adapter, MikroTik R52Hn, that is based on Atheros AR9220 chipset that is well supported by the open source Linux driver Ath9k [9]. Each interface is connected to a pair of external high gain antennas. The implementation is based on the Linux wireless modules [8].

The testbed were deployed in the second floor of the Bobby Chain Science Technology Building on campus with a floor plan as shown in Fig. 2. The building consists of laboratories, offices and classrooms. Both infrastructure (WLAN) and infrastructureless (Ad-hoc) modes were evaluated. In the infrastructure mode, one of the nodes was configured to work as the access point represented by the black triangle in Fig. 2 and two client nodes were placed at the locations A and B depicted as circles. The five hexagons in Fig. 2 show the infrastructure-less mesh topology. We conducted all our experiments at midnight to minimize the external surrounding interference such as walking people because all classes were dismissed and people were away from work. There are two non-overlapping channels (1, 6, 11) available in IEEE 802.11. We chose channel 1 since channel 6 and channel 11 were heavily used by surrounding networks. As for the software tools, we use *iperf* to generate and collect UDP traffic from clients to the access point. We did not choose TCP traffic because we would focus on the evaluation of the MAC access performance. The network was set in the IEEE 802.11n mode with two transmission streams and 20MHz bandwidth. With this configuration, the bit rate set consists of 13, 26, 39, 52, 78, 104, 117, and 130 Mbps. In our experiments, the basic rate was 13 Mbps when there were two streams enabled over 20 MHz bandwidth with short SGI disabled in IEEE 802.11n. In experiments, α in Eq. (1) was set to 100 and the weighted interval t_w for temporal fairness was set to 100 ms.

The traffic transmission lasted for 20 seconds for each run of experiment. The *iperf* manual reports that 10 seconds are enough to measure the network throughput. In order to eliminate the variation at the startup and completion of the experiment, we trimmed 5-second data at the beginning and the end of each measurement, so there are still 10 seconds of data. We repeated each scenario six times and the final data in the figures were averaged from the multiple runs.

2) *Infrastructure Mode*: This experiment is designed to evaluate the performance of the proposed *overlapped con-*

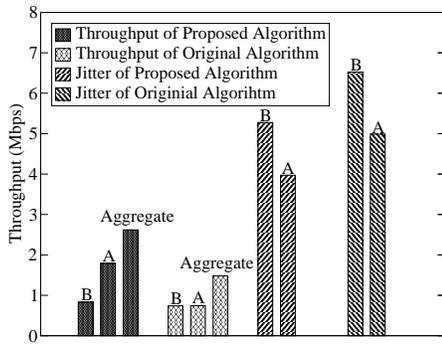


Fig. 3: Infrastructure Mode

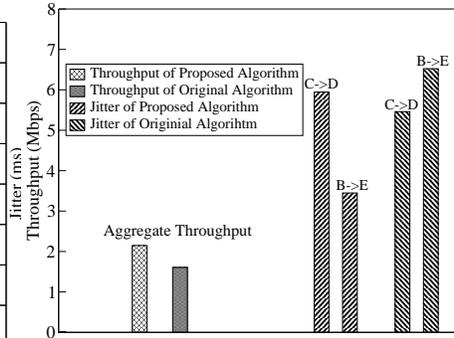


Fig. 4: Ad-hoc Mode

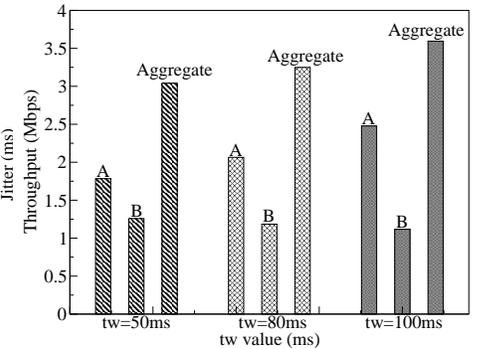


Fig. 5: Impact of t_w on Throughput

tion in infrastructure mode with an access point and two client nodes as in Fig. 2. *A* supported 130 Mbps and *B* had 52 Mbps. The transmissions were from the clients to the access point. In this experiment, the access point was configured to run the *iperf* server to receive the UDP traffic from clients.

In this experiment, we measured the jitter and throughput of UDP packets reported by the *iperf* server. Jitter is defined as the latency variations among packets. The results are shown in Fig. 3. The left two plots respectively show the throughput of the proposed algorithm and the original algorithm in CSMA/CA. The right two plots illustrate the jitter performance of these two algorithms. Two obvious observations are: 1) the proposed opportunistic access algorithm outperforms the original one in both throughput and jitter in general, because the opportunistic access spent much less time on backoff with smaller contention windows than the original access when nodes were at bit rates higher than the basic rate, 2) although the clients *A* and *B* have identical throughput in the original algorithm as expected, the opportunistic access enables them having different throughput based on their channel conditions and the aggregate network throughput is thereof significantly improved, by about 73% in the measured case.

3) *Ad-hoc Mode*: We also conducted performance evaluation on an ad-hoc topology as shown by the placement of the hexagons in Fig. 2. In our experiments, all nodes used the default MAC module, MAC 80211, in the Linux kernel. MAC 80211 is the wireless management layer in the Linux kernel. Two traffic flows went from *C* to *D*, and from *B* to *E*. *C* and *D* were so placed that they cannot hear each other and their flow had to be routed by *A*. The hops from *C* to *D* had bit rate of 52 Mbps and the hop from *B* to *E* was at 130 Mbps. As a result, the experimental topology resulted in a two-hop mesh network. In the experiments, the *iperf* flows started at *B* and *C* at the same time to transmit UDP traffic to *D* and *E* respectively. As in infrastructure topology, the jitter and throughput were measured.

Fig. 4 shows the measurement results. The left part of the figure illustrates the aggregate network throughput of the proposed opportunistic access and the original algorithm in CSMA/CA. It shows about 30% improvement of the throughput in this specific case. The right two plots on the figure depict the jitters of these two flows. As we can observe, when the opportunistic access was enabled, the jitter of flow *C* → *D*

only slightly increased, but that of flow *B* → *E* significantly reduced because the channel condition of *B* → *E* was better than that of *C* → *D* and therefore transmitted more frequently. In summary, the average jitter of proposed scheme is smaller than the original CSMA/CA.

4) *Impact of t_w* : As discussed in Section IV-C, the size of the weighted interval, t_w , has a significant impact on the network performance. Actually, it determines the degree of opportunism: if t_w is large, the average throughput will not be updated for a long period, and then the node with good channel conditions can always opportunistically access the channel. But this is at the cost of delaying the nodes with poor conditions and may hurt real-time applications that require small delay and jitters. We evaluated the impact of t_w on the throughput and jitter in the infrastructure mode with two clients: *A* at 130 Mbps and *B* at 39 Mbps. t_w took the values of 50, 80, and 100 ms in the evaluation.

Fig. 5 shows the impact on throughput. When t_w was doubled from 50 to 100 ms, the throughput of *A* improved by about 40% while that of *B* reduced by about only 15% and the aggregate network throughput improved by about 20%. This is because, as the opportunism is increased along with t_w , *A* got more chances than *B* in using the channel.

Fig. 6 shows the impact on jitter. The impact is opposite to that on throughput. When t_w was doubled, the jitter of *A* was reduced from 5 ms to about 3.5 ms while the jitter of *B* increased from 7.7 ms to 10 ms just because *A* got more chance and *B* got less chance of using the channel. However, both of their jitters are much lower than what is required for real-time applications.

B. Evaluation with Simulation

We developed all two proposed algorithms into the simulator *NS3* and comprehensively evaluated the performance with extensive simulations. Our algorithms were compared with the default CSMA/CA without opportunism for each case. The evaluation began with a simple infrastructure topology having one access point and two clients, then studied the impact of hidden terminal on opportunistic access, compared with opportunistic transmission in the case of mobility, and finally evaluated the performance on a multiple-hop mesh network. All experiments were performed with constant bit rate (CBR) UDP traffic at a rate of 10 Mbps for 5 minutes with packet

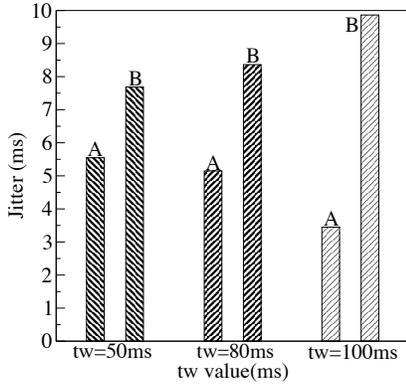


Fig. 6: Impact of t_w on Jitter

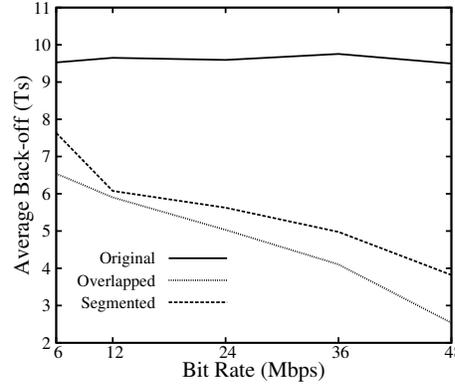


Fig. 7: Average Backoff

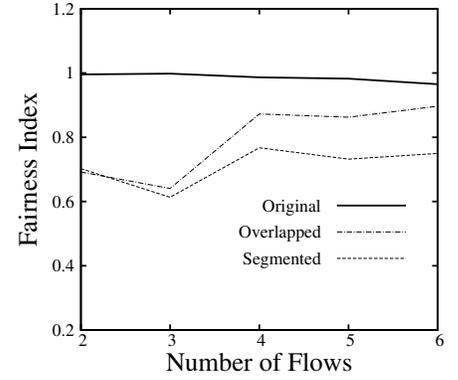


Fig. 8: Fairness

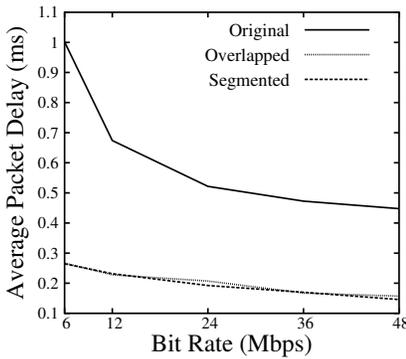


Fig. 9: Average Delay

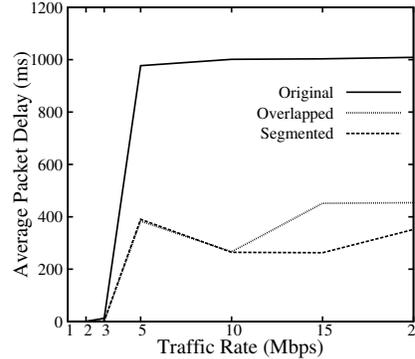


Fig. 10: Delay vs Traffic Rate

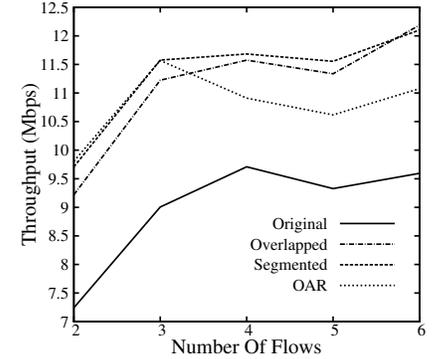


Fig. 11: Throughput under Mobility

size of 1K bytes. The results are averaged over 50 runs for each case. In simulations, the weighted interval t_w was conservatively set to 50 ms. In the following performance figures, “Original” represents the original CSMA/CA, “Overlapped” for the *overlapped contention*, “Segmented” for the *segmented contention*.

1) *Triple-node Infrastructure Topology*: We first evaluated the throughput and jitter performance of opportunistic access over the original algorithm in a simple topology: one access point and two client nodes. All nodes were in radio range of each other. These client nodes transmitted packets to the access point at different bit rates. One client node was set to a 54 Mbps, but the other node changed its bit rate from the IEEE 802.11 rate set consisting of 6, 12, 24, 36, and 48 Mbps.

2) *Average Backoff Time*: To further investigate how the opportunistic access improves the throughput and jitter, we collected the backoff time in terms of time slots of each transmission. Fig. 7 shows the average backoff time per successful transmission for each access algorithm. Average backoff slot time per packet is calculated as the sum of individual packet backoff time slots over the total number of packets transmitted successfully. From the figure, the opportunistic access algorithms consume much less time in backoff than the original access (up to 400% less at 48 Mbps). The overall network performance improves because 1) the equal probability access in the original algorithm results in almost constant average backoff, and 2) the opportunistic access algorithms spend less time on contention.

3) *Fairness Index Measurement*: Jain Fairness Index is the most commonly used metric to measure the fairness variation using the individual flow throughput over wireless networks. Basically Jains Fairness rates the individual throughput variation with respect to number of flows. If resources are allocated equally, then Jain fairness index will have the maximum value 1 and vice-versa. Jain Fairness Index is given by:

$$I_J = \frac{(\sum_{i=1}^n r_i)^2}{n \sum_{i=1}^n r_i^2} \quad (12)$$

Where r_i and n are the allocated resources to node i and the total number of nodes respectively.

Fig. 8 shows that the the fairness of opportunistic access scheme gets increased as the number of flows increases. This is because the opportunistic access actually reduces the contention time of high bit rate nodes, thereby giving opportunism to send more packets but the lower bit rate node remain in its regular channel access time. The fairness of opportunistic transmission is also compared with opportunistic access which shows that the resources are allocated equally for long run.

4) *Delay Performance*: Fig. 9 plots the measured average delay at different bit rates. The opportunistic access shows a significant improvement in delay as compared to the original access. This is because the high bit rate nodes in opportunistic access has very less contention time and transmits packets rapidly. The lower bit rate nodes experiences a much larger delay comparatively because of collision due to back-off terminate at the same time and the contention window size gets doubled every time when it encounters a collision. Yet the overall network performance in delay is improved significantly

over the default CSMA/CA method. To show more clearly, we further studied the delay performance with increase in traffic rate Fig. 10. As expected, the delay gets increased as CBR traffic rate increases because more packets are transmitted frequently. Although the variation in individual packet delay between high bit rate nodes and low bit rate nodes is high, the overall performance of opportunistic access is improved.

5) *Comparison with Opportunistic Transmission under Mobility and Auto Rate:* We also evaluated the performance in an infrastructure topology of multiple flows with mobility. We tested topologies with a different number of nodes varying from 2 to 6 nodes with one flow between each node and the access point. The maximum transmission range of a node was set to 30m and mobility was enabled such that each node would be moving randomly within the bounded area of 50m×50m at a speed of 5 m/s. The nodes sometimes move out of range and packet losses occur more frequently. Because of mobility, the supported bit rate has to be dynamically adapted. *RRAA* [19] rate adaptation algorithm was enabled for this purpose. In addition, we also implemented Opportunistic Auto Rate (*OAR*) protocol as a representative of *opportunistic transmission* algorithms to compare with the opportunistic access in this experiment setting.

Fig. 11 shows the throughput performance of the opportunistic access algorithms, the original access and *OAR* with mobility and *RRAA* in the case of multiple flows. The *X*-axis represents the number of flows and the *Y*-axis represents the throughput for the entire network. From the figure, the opportunistic access (our algorithms) and the opportunistic transmission (*OAR*) have close performance when the network only has a couple of nodes, but the opportunistic access gradually outperforms the opportunistic transmission as the number of mobile nodes increases. This is because of the difference in the opportunism between access and transmission. With the growing number of mobile nodes, the node diversity increases as well. *OAR* does not exploit node diversity and it uniformly selects a node to transmit. As a result, it does not favor the node with the best channel condition to use the channel each time. On the contrary, opportunistic access exploits node diversity. With more nodes, it is more likely that someone is at a very high bit rate. Because the opportunistic access always favors the node with the highest bit rate to use the channel, the overall network performance is improved.

VI. CONCLUSION

IoT requires effective medium access protocols for wireless communication. This work proposes two opportunistic random access variants to exploit node diversity in wireless networks. These algorithms enable nodes to access the shared wireless channel based on their channel conditions so that the node at the highest achievable bit rate is favored. To avoid starving nodes with poor channel conditions, a slow filtering scheme is proposed to maintain temporal fairness among nodes. With extensive experiments on a developed Linux-based testbed and the *NS3* network simulator, the proposed opportunistic access schemes significantly improve the network performance in throughput, delay, and jitter, which can offer significant advantages for supporting future IoT applications..

VII. ACKNOWLEDGMENTS

The authors would thank the support from the National Science Foundation through the awards #1041292 and #1041095 to perform this work. Gordon Petty helped coding.

REFERENCES

- [1] B. Sadeghi, V. Kanodia, A. Sabharwal, E. Knightly, Opportunistic media access for multirate ad hoc networks, *MobiCom* 2002
- [2] B. Sadeghi, V. Kanodia, A. Sabharwal, and E. Knightly. *OAR: An opportunistic auto-rate media access protocol for ad hoc networks*. *Wireless Networks*, 11:39-53, 2005
- [3] V. Kanodia, A. Sabharwal, and E. Knightly, *MOAR: a multi-channel opportunistic auto-rate media access protocol for ad hoc networks*. In *proceedings. First International Conference on Broadband Networks 2004*
- [4] Vaduvur Bharghavan, Alan Demers, Scott Shenker, Lixia Zhang. *MACAW: a media access protocol for wireless LAN's*. *SIGCOMM 1994*
- [5] G. Holland, N. Vaidya, P. Bahl. *A rate-adaptive MAC protocol for multi-Hop wireless networks*. *MobiCom* 2001
- [6] *IEEE LAN MAN Standards*, part 11: *Wireless LAN medium access control (mac) and physical layer (phy) specifications*, March 1999.
- [7] Martin Heusse, Franck Rousseau, Gilles Berger-Sabbatel, Andrzej Duda, *Performance Anomaly of 802.11b*, *Infocom* 2003
- [8] Vipin M, Srikanth S, *Analysis of Open Source Drivers for IEEE 802.11 WLANs*, *ICWCSC* 2010
- [9] <http://linuxwireless.org/en/nodes/Drivers/ath9k>
- [10] H. Zhai, Y. Fang, M.C. Yuang, *Opportunistic Media Access Control and Rate Adaptation for Wireless Ad Hoc Networks*, *ICC*, 2004
- [11] Hiroyuki Yomo, Petar Popovski, *Opportunistic Scheduling for Wireless Network Coding*, *ICC*, 2007
- [12] Xiaojun Liu, Edwin K. P. Chong, Ness B. Shroff, *Opportunistic transmission Scheduling with resources-sharing constraints in wireless networks*, *JSAC*, vol. 19, no. 10, pp. 2053-2064, 2001
- [13] Yonghe Liu, Edward W. Knightly, *Opportunistic Fair Scheduling over Multiple Wireless Channels*, *Infocom* 2003
- [14] Sanjit Biswas, Robert Morris, *Opportunistic routing in multi-hop wireless networks*, *CCR*, vol. 34, no. 1, pp. 69-74, 2004
- [15] Sanjit Biswas, Robert Morris, *ExOR: Opportunistic multi-hop routing for wireless networks*, *SIGCOMM*, 2005
- [16] Szymon Chachulski, Michael Jennings, Sachin Katti, Dina Katabi, *Trading structure for randomness in wireless Opportunistic routing*, *SIGCOMM* 2007
- [17] Luciana Pelusi, Andrea Passarella, Marco Conti, *Opportunistic networking: data forwarding in disconnected mobile ad hoc networks*, *IEEE Communications Magazine*, vol. 44, no.11, pp. 134-141. 2006
- [18] Godfrey Tan, John V. Guttag. *Time-based Fairness Improves Performance in Multi-Rate WLANs*. *USENIX* 2004
- [19] Starsky H. Y. Wong, Hao Yang, Songwu Lu, and Vaduvur Bharghavan. "Robust rate adaptation for 802.11 wireless networks." In *Proceedings of the 12th annual international conference on Mobile computing and networking (MobiCom '06)*. ACM, New York, NY, USA, 2006
- [20] J. Wang, H. Zhai, Y. Fang and M.C. Yuang. *Opportunistic media access control and rate adaptation for wireless ad hoc networks*. *IEEE International Conference on Communications*, 2004
- [21] Vaidya, N.; Dugar, A.; Gupta, S.; Bahl, P.; "Distributed fair scheduling in a wireless LAN;" *IEEE Transactions on Mobile Computing*, vol.4, no.6, pp. 616- 629, Nov.-Dec. 2005
- [22] N. Vaidya, P. Bahl, and S. Gupta. *Distributed fair scheduling in a wireless LAN*. In *Proceedings of the 6th annual international conference on Mobile computing and networking (MobiCom '00)*, New York, NY, USA, 2000
- [23] Chan-soo Hwang; Cioffi, J.M.; , "Opportunistic CSMA/CA for achieving multi-node diversity in wireless LAN;" *IEEE Transactions on Wireless Communications*, vol.8, no.6, pp.2972-2982, June 2009