

RESEARCH

Open Access



# N-DCF: MAC overhead reduction using narrow channel contention in wireless networks

Jungmin So and Ayinebyona Eliab\*

## Abstract

This paper presents a medium access control protocol for wireless networks that improves system spectral efficiency using multiple channels. Use of multiple narrow channels has already been proposed as a solution to reduce MAC overhead, but merely dividing a wide channel into narrow channels cannot mitigate the negative effect of packet collisions especially when the network is densely populated. The proposed protocol, called N-DCF, adds protocol support to reduce packet collisions and improve channel utilization. The main idea is to let each node contend on one of the multiple channels and give the node privilege to access other channels when it successfully finishes transmission on its contention channel. When a node gains privilege to a channel, it can access the channel without contention. Privilege is gained probabilistically so that other nodes can have chance to win the channel and obtain privilege for other channels as well. N-DCF reduces channel idle time by conducting random backoff only on one of the narrow channels and reduces packet collisions by assigning contention channel randomly. Idle channels are not wasted because nodes can access other channels once the node wins its contention channel. Performance evaluation shows that the proposed scheme achieves higher throughput than 802.11 DCF without harming the fairness, regardless of node density, packet size, and number of radios.

**Keywords:** Wireless LANs, MAC overhead reduction, Narrow channels

## 1 Introduction

IEEE 802.11 DCF, a medium access control used in current-day wireless LANs, uses random backoff mechanism to mitigate packet collisions [1]. Before transmitting a packet, a node picks a random number from a range called contention window and waits for a duration of time according to the selected number. The purpose of random backoff is to have a single node begin transmission, although it is still possible that multiple nodes transmit simultaneously, resulting in a packet collision. Once a node starts transmitting, other nodes sense the channel as busy and wait until the channel becomes idle again.

The overhead of this random access protocol consists of channel idle time (duration of time when all nodes are waiting and no one is using the channel) and loss from packet collisions. There is a trade-off between average

channel idle time and probability of packet collision. If the contention window is too small, chance of packet collision increases. If the contention window is too large, channel idle time increases. An optimal contention window size depends on factors such as number of contending stations and average packet size.

There are other approaches for medium access control which do not rely on randomness, such as token-passing MAC [2]. While these protocols do not require channel idle time, it has its share of challenges and costs such as handling node join and leave, loss of token, and contention between token-holders of neighboring networks. Random access protocols do not have these problems and are generally much simpler to implement.

In this paper, we propose a protocol called N-DCF (Narrow DCF) which reduces cost of random access protocols using multiple channels. We keep the random access approach as a baseline and add privileged access similar in order to reduce MAC overhead. Our assumption is that every node is capable of independently accessing multiple

\*Correspondence: [eliabic@hallym.ac.kr](mailto:eliabic@hallym.ac.kr)  
Smart Computing Lab. and Dept. of CE, Hallym University, Chuncheon 24252, South Korea

channels. For example, a node can transmit on a 20 MHz channel while receiving on another 20 MHz channel. This assumption can be realized by equipping multiple network interfaces on each node. Also, it would be feasible to build a single network interface that is capable of accessing multiple narrow channels [3]. The requirements for implementing the proposed scheme are described in Section 4. We do not assume full-duplex capability.

IEEE 802.11 standard specifies channels with various widths, such as 20 MHz, 40 MHz, 80 MHz, and 160 MHz. It is also possible to use 5 MHz and 10 MHz channels. Even if nodes can access multiple channels simultaneously, using multiple narrow channels instead of a single wide channel comes with a cost due to the guard band. Also, using a very narrow channel such as a 5MHz or a 10MHz channel requires longer time slots due to increased preamble transmission time. However, these losses are compensated by reduced MAC overhead. More specifically, channel idle time and number of packet collisions can be reduced through “narrow channel contention.” The idea is to have each node contend for one of the many available channels. Once the node wins the channel, it gains privilege to access other channels without contention. To avoid starvation, privileges are gained probabilistically. As shown in the performance evaluation, N-DCF achieves higher throughput compared to the conventional 802.11 DCF.

The rest of the paper is organized as follows. Section 3 discusses the current state of the WLAN standards and motivates the problem. Section 4 describes the proposed protocol, N-DCF, which uses narrow channel contention and probabilistic privileged access to improve system spectral efficiency. Section 5.1 shows the performance of N-DCF, compared with other schemes and configurations. Section 6 discusses existing work relevant to the goal of this paper. Finally, Section 7 concludes the paper with remarks for future work.

## 2 Methods

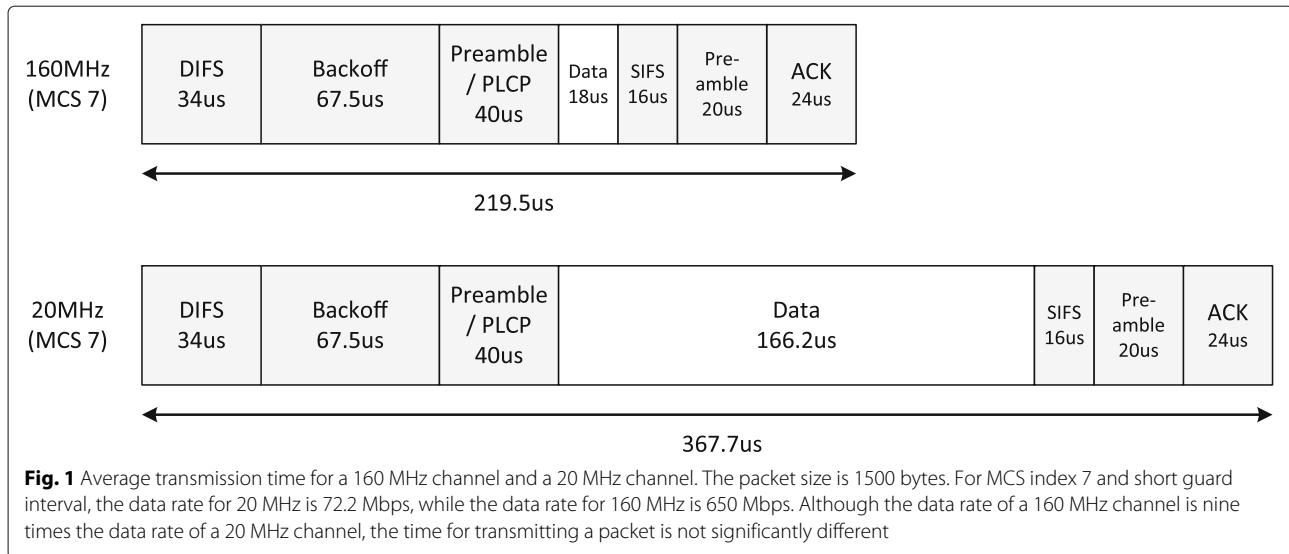
This paper proposes a new MAC protocol for wireless networks, which uses multiple interfaces cooperatively in order to reduce channel idle time and packet collision, and thus achieve better system throughput. The performance of the proposed protocol was evaluated through extensive simulations. The ns-3 simulator was used for simulation, which is widely used for network protocol evaluations. The proposed protocol was implemented as a new module for the simulator. Performance of the protocol was compared with 802.11 DCF which is the protocol currently used in wireless LANs. Throughput and fairness were measured, where Jain’s fairness index was used as the metric of fairness. Impact of varying number of nodes, number of radios, and probability of gaining privilege were studied in both WLAN and ad hoc network environments.

## 3 Preliminaries

The recent wireless LAN standard IEEE 802.11ac supports up to 160 MHz channels in the 5 GHz band. For example, channel 50 is a 160 MHz channel which ranges from 5170 to 5330 MHz. Eight 20 MHz channels fit in that range, numbered 36, 40, 44, 48, 52, 56, 60, and 64. If a mobile node and its associated AP are capable of accessing eight 20 MHz channels simultaneously, they can make a choice between using a single 160 MHz channel and eight 20 MHz channels. Without considering MAC overhead, it is beneficial to use a single 160 MHz channel because it has 468 data subcarriers whereas a 20 MHz channel has only 52 data subcarriers. However, using the wide channel does not necessarily achieve higher throughput, because the relative cost of MAC overhead can be higher for high rate channels.

Suppose a node wants to send a packet. After the channel becomes idle, a node has to wait for DIFS and then count down the backoff counter. DIFS is  $34 \mu s$ , and the average backoff time is  $67.5 \mu s$  provided that the channel is constantly idle while the node is counting down the backoff counter. Once the node gains access to the channel, it first sends a preamble followed by data. The preamble transmission time for a data packet is  $(36 + 4N) \mu s$ , where  $N$  is the number of spatial streams. When the destination node receives the packet, it waits for SIFS ( $16 \mu s$ ) and sends an ACK which also includes sending the preamble. The ACK is sent using the legacy mode, and so its preamble is  $20 \mu s$  long. The time for sending data and ACK will depend on their size and data rate. It is important to note that while transmission time for data decreases with increased data rate, the MAC overhead stays the same (assuming the time slot duration is the same). Figure 1 shows the time required to transmit a 1500-byte packet through a 20 MHz channel and a 160 MHz channel. For MCS index 7 and short guard interval (SGI), the data rate for 20 MHz is 72.2 Mbps, while the data rate for 160 MHz is 650 Mbps. The data rate of a 160 MHz channel is nine times the data rate of a 20 MHz channel. However, the average time for sending a packet is  $367.7 \mu s$  and  $219.5 \mu s$  for a 20 MHz channel and a 160 MHz channel, respectively. The packet sending time of a 20 MHz channel is less than twice the packet sending time of a 160 MHz channel.

In order to mitigate the effect of MAC overhead, recent standards allow frame aggregation. Instead of sending a single packet, multiple packets are sent after acquiring the channel. Block ACKs (BA) are used to acknowledge the multiple packets. Frame aggregation basically increases the data portion in the transmission time, thereby reducing the portion of MAC overhead. The maximum payload size is the amount of data that could be sent in 5.484 ms. At the highest data rate in 802.11ac, the payload size could be up to 4,692,480 bytes. Since the time required



for channel contention and ACK (refer to Fig. 1) stays fixed at approximately 200  $\mu$ s, the percentage of MAC overhead becomes minimal. However, using larger packets lead to higher loss when there is a packet collision, since the whole packet is lost.

Figures 2 and 3 show results from a simple experiment using the ns-3 simulator [4]. Figure 2a, b shows the average throughput when the packet size is varied. In this experiment, a single AP sends UDP traffic to a single mobile station. Figure 2a is the result for a single channel with different widths, whereas Fig. 2b is the result for the fixed aggregated channel width. When the packet size is small, using wider channel does not achieve significantly higher throughput. For a 1500-byte packet, the average throughput is 32 Mbps for a single 20 MHz channel and 54 Mbps for a single 160 MHz channel. This result corresponds to the packet transmission time described in Fig. 1. When the packet size becomes large using frame aggregation, throughput increases due to reduced MAC overhead, especially for the wider channels. When the packet size is 65,535 bytes, a 20 MHz achieves 66 Mbps of throughput where as a 160 MHz channel achieves near 500 Mbps of throughput. Figure 2b shows the average throughput for different channel configuration, when the aggregate bandwidth is the same. When we need to use small packet sizes such as for delay-sensitive applications, it is better to use multiple narrow channels instead of a single wide channel, provided that the hardware supports the configuration. However, if we can allow large packets, a single wide channel will achieve throughput comparable to multiple narrow channels.

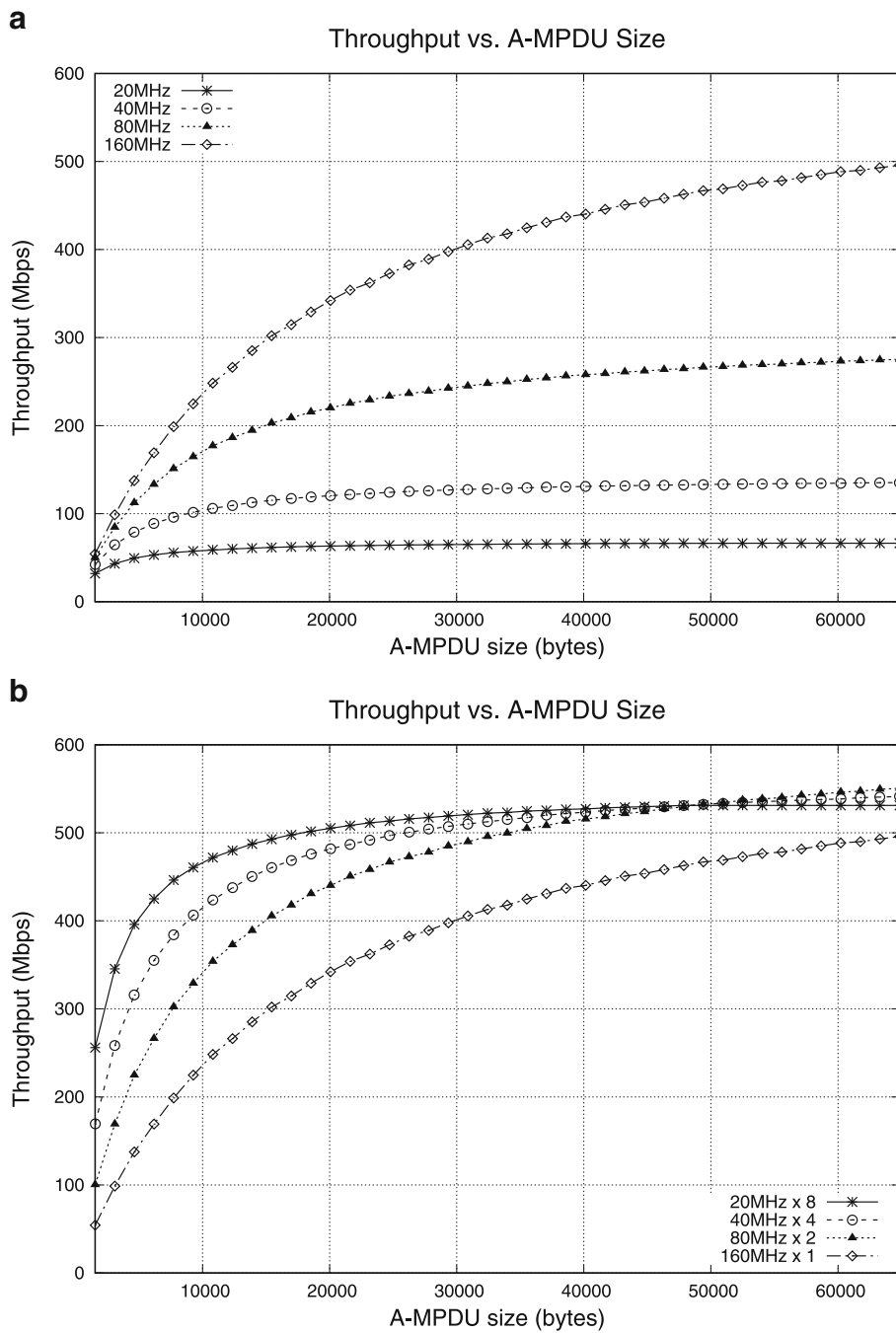
Figure 3a, b shows the average throughput when the node density is varied. Figure 3a is the case when eight 20 MHz channels are used, and Fig. 3b is the case when a single 160 MHz channel is used. The lines in each

graph are results for different packet sizes. Intuitively, the aggregate throughput degrades when the number of nodes increases due to packet collisions. Also, we can observe that throughput degradation is much more significant for large packets. For example, when we send 1500-byte packets through a single 160 MHz channel, the throughput drops from 54 to 32 Mbps when the number of nodes increases from 1 to 200. However, for a 65,535-byte packet, the throughput drops from 500 to 150 Mbps. Another observation we can make is that when the node density is high, using a single 160 MHz channel with frame aggregation achieves higher aggregate throughput than using eight 20 MHz channels.

In summary, dividing a single wide channel into multiple channels can achieve higher throughput, because the MAC overhead is relatively small for narrow channels [3]. However, using narrow channels does not automatically improve throughput especially when large packet sizes are allowed. Depending on packet size and node density, there are cases where it is better to use a single wide channel. It is possible to design schemes that dynamically choose channel widths based on channel and traffic conditions [5]. However, what is more important is to reduce the packet collision rate. Without additional protocol support, dividing channels and applying the basic 802.11 DCF protocol independently on all multiple channels will not decrease collision rate. The proposed scheme provides protocol support for multiple narrow channels in order to reduce packet collision rate while fully utilizing the available frequency resource.

#### 4 Proposed protocol

We assume all nodes in the network are capable of accessing all available channels simultaneously, either using multiple network interfaces or other hardware such as



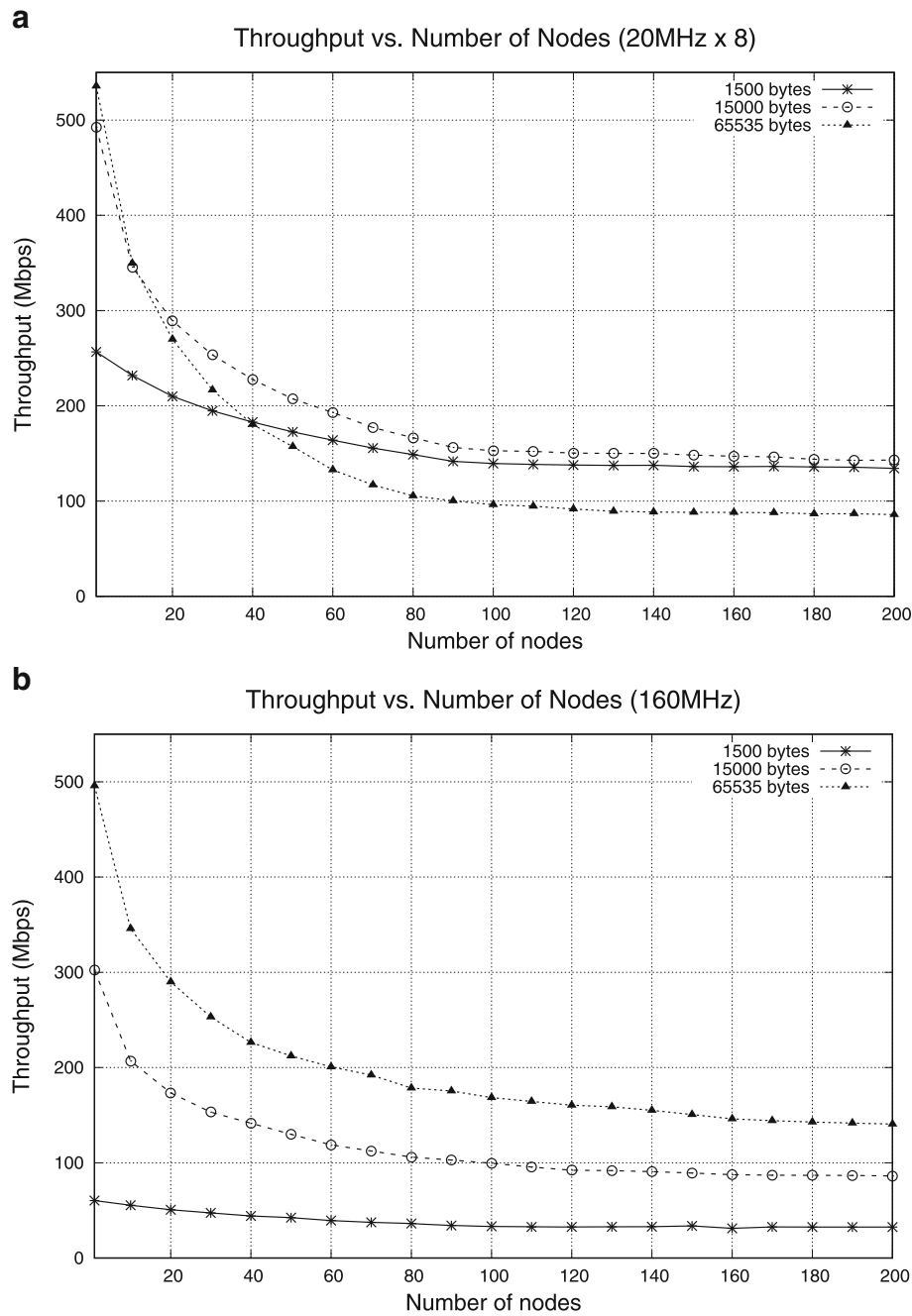
**Fig. 2** Average throughput varying packet size. **a** Single channel with different widths. **b** Fixed aggregate width (160 MHz)

compound radios [3]. The simplest method to reduce packet collisions is to distribute nodes across the available channels. More specifically, we can let each node randomly choose one of the channels and perform channel contention using the basic 802.11 DCF. The problem with this simple method is that every node will only use one of the available channels at a time, even if other channels are idle. If we allow a node to contend at multiple channels,

the collision rate will increase. The proposed protocol, N-DCE, makes each node contend on one of the channels and use that contention information to access other channels without paying further overhead.

**4.1 Privileged access**

Suppose there are two channels, channel 1 and channel 2. To transmit a packet, all nodes contend for channel 1 using



**Fig. 3** Average throughput varying node density. **a** 20 MHz x 8. **b** 160 MHz x 1

the random backoff scheme of 802.11 DCF. (In this case, channel 1 becomes the “contention channel.”) Suppose node D wins channel 1. Then, node D transmits its packet on the channel and receives an ACK from its receiver. When node D wins channel 1, we are going to give node D the privilege to access channel 2 without contention. Since node D has won channel 1, we can expect that node D will be the only privileged node in its neighborhood, and there will be no collision when node D transmits on channel 2.

The implicit assumption here is that the channels come from the same frequency band and thus have similar propagation characteristics. However, it is always possible that channel 2 is occupied when node D wins channel 1. Even with privilege, node D waits until the channel becomes idle and then starts its transmission.

One important question to ask is as follows: exactly when does a node acquire privilege, and when does the privilege end? The most basic approach is to give a node

privilege while the node is accessing the contention channel. This approach is illustrated in Fig. 4. Here, nodes D and E contend for channel 1. First, node D wins channel 1. While node D is transmitting, channel 2 becomes idle. Unprivileged nodes follow the 802.11 DCF procedure, in which a node waits for DIFS period and starts counting down its backoff counter. However, node D just waits for SIFS period and immediately accesses the channel without performing backoff. While node D is transmitting on channel 2, it finishes transmission on channel 1 and receives an ACK from the receiver. That is when the privilege is released from node D. Now, through contention, node E wins channel 1 and transmits its packet. If channel 2 becomes idle while E is transmitting, E now has the privilege to access channel 2 without contention. It is possible that when channel 2 becomes idle, no one is transmitting on channel 1. In that case, nodes gain access to channel 2 through normal contention. This scheme works well when there is no collision at the contention channel. The problem with this approach is that if two nodes access the contention channel at the same time thereby causing a packet collision, both nodes will gain privilege on other channels and cause collisions on those channels as well. Thus, this approach does not help much in reducing collision rate.

An alternative approach is to assign privilege to a node only when the node *successfully finishes* transmitting on the contention channel. N-DCF takes this approach, which is described in Fig. 5. On channel 1, node D gains access to the channel through normal contention. On winning the channel, node D transmits its packet and receives an ACK from its receiver. Once node D receives an ACK, the node acquires privilege to access channel 2. While possessing the privilege, node D does not perform backoff on channel 2. Once the channel becomes idle, node D accesses the channel right away after waiting for SIFS. (Node D still has to wait if the channel is busy, even with the privilege.) Since node D has successfully finished transmission on channel 1, it will be the only node in its neighborhood that has the privilege (assuming the transmission ranges are similar for the channels.)

This approach has potential to reduce packet collisions. However, compared with the previous approach, it is more

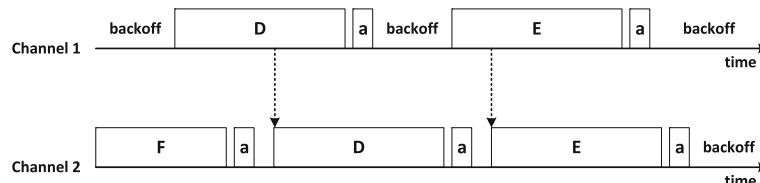
difficult for a node to figure out when to give up its privilege. After receiving the ACK, node D gains privilege for channel 2. Node D should release this privilege once another node successfully finishes transmission on channel 1. So when node D overhears an ACK on channel 1, it should give up its privilege on other channels. However, not every node can always overhear ACK and release its privilege at the right time, such as a scenario shown in Fig. 6. In the figure, node A first wins the contention channel and transmits its packet to C. When it receives an ACK from C, it gets the privilege for the other channels. Later, node B wins the contention channel and sends a packet to D. When B receives ACK from D, it will obtain the privilege. When it happens, node A should overhear the ACK and release its privilege. However, node A is too far from node D and thus cannot overhear the ACK. Thus, node A does not release its privilege, resulting in a scenario where multiple nodes possess privilege. This is a disastrous scenario, because packet collision is guaranteed at other channels.

In order to prevent multiple privileged nodes in a neighborhood, a node has to release its privilege even if it does not overhear an ACK, if the receiver of the current transmission is not its neighbor. For this purpose, each node maintains a list of neighbors. A new neighbor is added to the list whenever a node overhears a data packet or an ACK. A neighbor node is deleted from the list if the node is not seen for a certain amount of time. Based on the neighbor list, a privileged node releases its privilege upon the following conditions:

1. If the node overhears an ACK, it releases its privilege when it finishes overhearing the ACK.
2. If the node overhears a data packet and its destination is not in the neighbor list, the node releases its privilege when it finishes overhearing the packet.

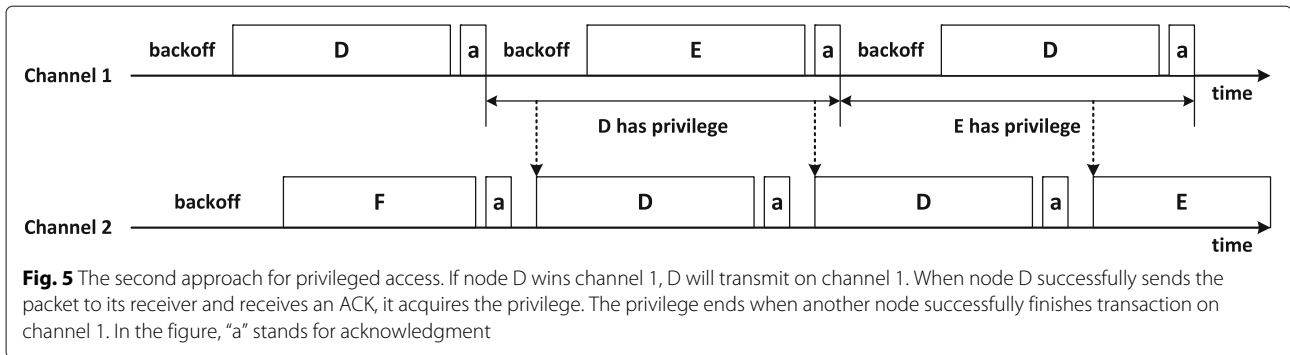
#### 4.2 Privilege assignment and probabilistic access

In the privileged access mechanism described above, there is a contention channel where every node needs to gain access in order to obtain privilege for other channels. Suppose we have  $c$  channels, channel 1 through channel  $c$ . Then, we can designate channel 1 as the contention chan-



**Fig. 4** The first approach for privileged access. When node D wins channel 1, it acquires privilege to access channel 2 without contention. The privilege is taken away when node D receives an ACK, or the ACK timer expires. In the figure, "a" stands for acknowledgment





nel where all nodes compete for the channel. The node who wins the contention channel gains privilege for other channels. Unprivileged node may access other channels as well, but only through normal contention. Since the privileged node preemptively accesses other channels, other nodes have little probability of accessing those channels.

The privileged access mechanism removes most of contention overhead in the channels other than the contention channel. However, still all nodes need to contend for the contention channel. When node density is high, lots of collisions may occur in the contention channel. As discussed earlier, with multiple channels, we can potentially distribute nodes across channels and reduce probability of packet collision. So instead of designating one of the channels as a common contention channel, N-DCF lets each node randomly choose a channel to contend. Moreover, a node may change its contending channel for every packet.

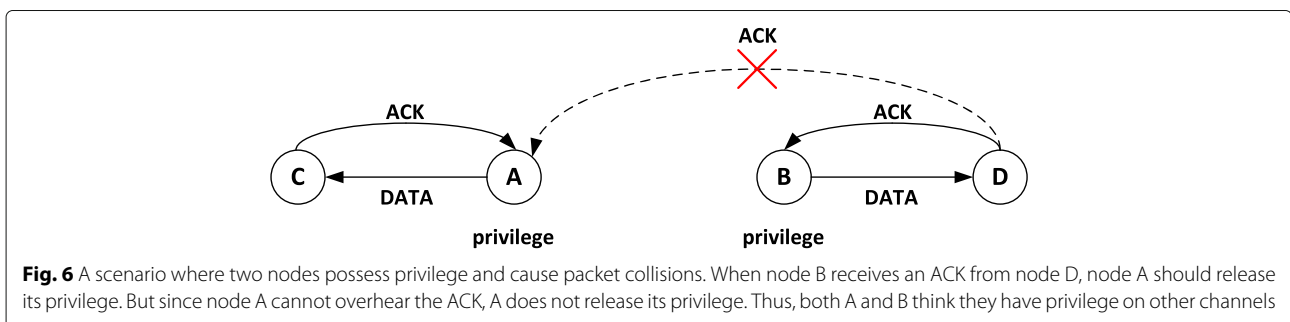
When we allow selecting contention channel arbitrarily, multiple nodes may acquire privilege from successfully transmitting their packets on their respective contention channels. For example, node D chooses channel 1 and node E chooses channel 2 as their contention channels. If they successfully transmit packets on their contention channels, they both acquire privileges for accessing other channels. If both of the privileged nodes transmit packets on channel 3, there will be a collision.

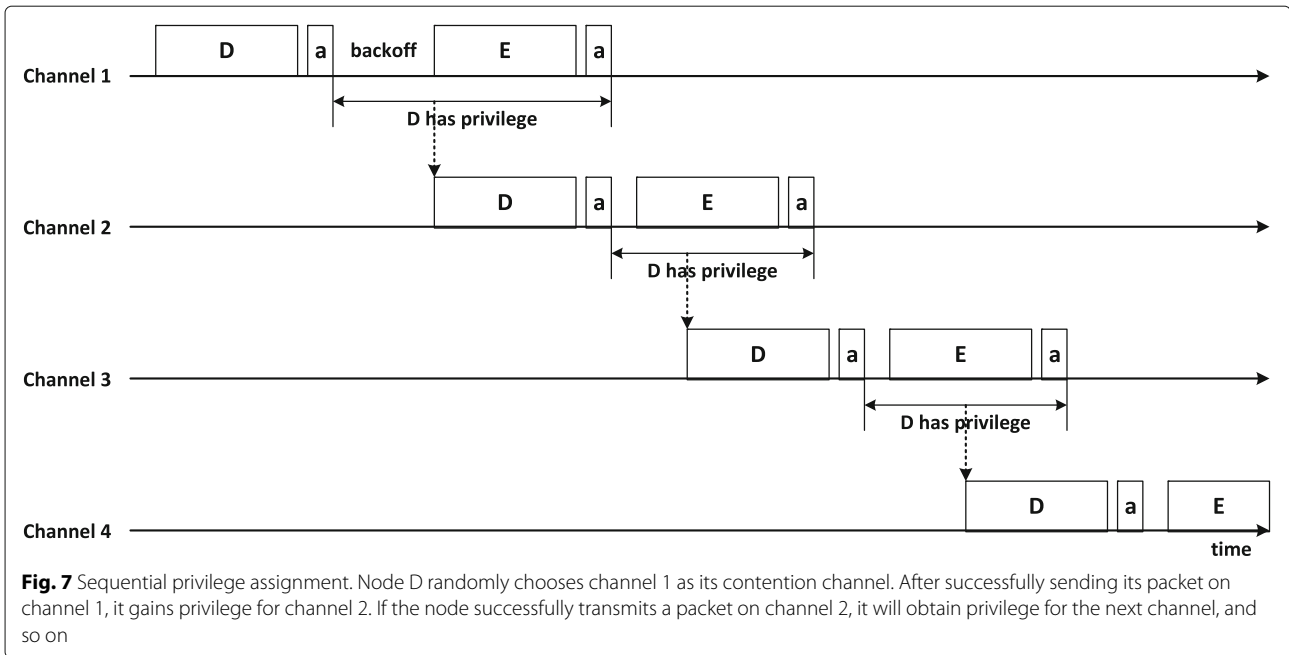
Instead of assigning privilege for all other channels, N-DCF manages separate privilege for each channel.

Suppose there are  $c$  channels. To transmit a packet, node D chooses channel  $i$  for contention. Once node D sends a packet and receives an ACK on channel  $i$ , it gains privilege for channel  $i + 1$ . (If the contention channel is channel  $c$ , the obtained privilege is for channel 1.) If node D successfully sends a packet on channel  $i + 1$ , it acquires privilege for channel  $i + 2$ . Eventually, a single node may obtain privilege for all channels, although at different times. We call this behavior “sequential privilege assignment,” and it is illustrated in Fig. 7.

Using this scheme, the number of contending nodes reduces for each channel and thus collision rate decreases. However, still one problem is left. Suppose node D wins channel  $i$  before other nodes win their respective contention channels. After receiving ACK, it will access channel  $i + 1$  without contention. As described earlier, once the channel becomes idle, the privileged node waits for SIFS before transmitting its packet, whereas other non-privileged nodes need to wait for DIFS plus remaining backoff time. So when node D gains privilege for channel  $i + 1$ , all nodes that were contending on the channel will need to defer because of node D. It is possible that this kind of phenomenon continues to happen, and some nodes may result in starvation.

In order to promote fair share of channel resource, we use probabilistic privileged access, where privilege is gained with probability  $p$  which is a tunable parameter. Suppose node D wins channel  $i$  and successfully sends a packet there. Then, the node randomly picks a number





between 0 and 1. If the picked number is less than  $p$ , the node gains privilege for channel  $i + 1$ . If the number is larger than  $p$ , the node still can access channel  $i + 1$ , but it should follow the normal random backoff procedure and contend with other nodes for the channel. If  $p$  is large, higher chance is given to privileged nodes. If  $p$  is small, higher chance is given to contending nodes rather than the privileged node. With large  $p$ , the overall system throughput will be higher because most channel access will be without contention. However, fairness can be low, because a few preoccupied nodes continue to have privileged access on channels. On the contrary, with small  $p$ , fairness will be higher but overall system throughput will be lower.

### 4.3 Implementation details

In order to implement N-DCF, several modifications should be made to the radio structure. Traditionally, a network interface has a packet queue, plus MAC and PHY modules. Packets to be transmitted are first stored in the interface queue. Then, the MAC starts contending for the channel. When channel access is granted, the packet is transmitted from the interface. If a node has multiple network interfaces, we need a packet scheduler who maps packets to interfaces [6]. Each network interface has queues, so packets are assigned to interfaces before contending for the channel.

However, in N-DCF, a node selects one of the channels and contends only on that channel. Once the node wins the contention channel, it has chance to obtain privilege for the next channel. If the node sends a packet on that channel, it may gain privilege for the next channel, and so

on. If packets are sent to the packet queue of each interface, significant out-of-order delivery could happen due to protocol behavior. It is also problematic if the packets are only sent to the interface tuned to the contention channel. Then, once a node gains privilege for the next channel, there is no packet to send in the packet queue of that channel.

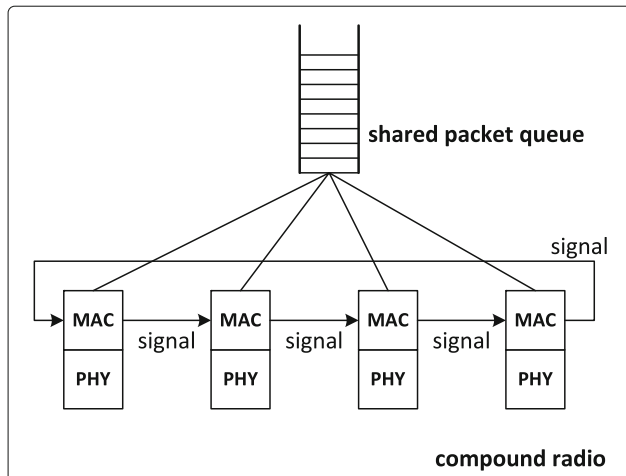
To address this problem, N-DCF maintains a single shared queue for all radios in the compound radio. When a packet is inserted to the shared packet queue, the MAC of the contending channel starts contending for the channel. Once the radio gains channel access, a packet is pulled from the shared packet queue and transmitted. Once the radio receives an ACK and gains privilege to the next channel, the MAC of contending channel signals the MAC of the next channel to pull a packet from the queue and send it immediately. The structure of the compound radio is shown in Fig. 8.

## 5 Performance evaluation

### 5.1 Simulation setup

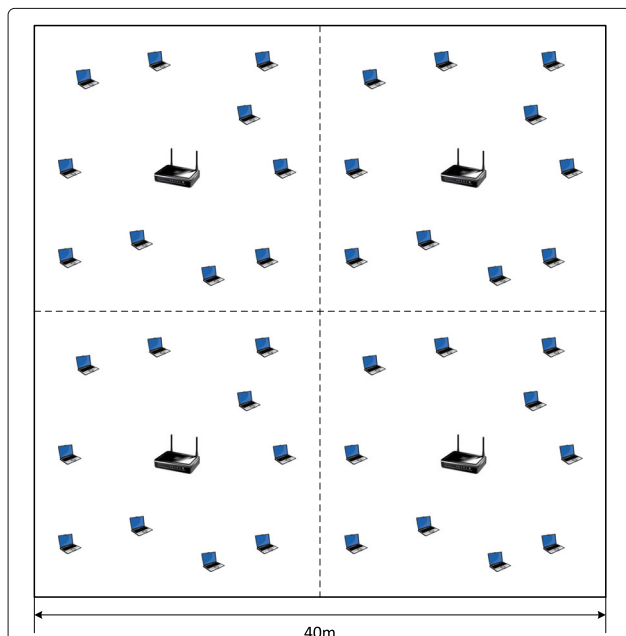
We have evaluated performance of N-DCF through simulations using the ns-3 simulator [4]. The compound radio module was implemented which consists of multiple MAC-PHY pairs and a shared packet queue. The MAC modules of interfaces are connected with each other to signal messages related to privileges. Each MAC-PHY pair operates according to the 802.11ac standard. The simulation environment is a multi-WLAN environment, illustrated in Fig. 9. To create the environment, four APs were placed at the center of each 20 m × 20 m square area, and mobile stations were randomly deployed in the





**Fig. 8** Structure of compound radio used to implement the proposed scheme. A MAC-PHY pair operates for each narrow channel. Each MAC runs 802.11 DCF, but does not have its own packet queue. Instead, when the channel access is granted, the MAC dequeues a packet from the shared packet queue and transmits the packet. Neighboring MACs are connected so that when a node successfully receives an ACK, the MAC of channel  $i$  can signal the MAC of channel  $i + 1$  to perform privileged access

40 m × 40 m region covering four square-shaped cells. The transmission range is approximately 30 m for packets using MCS (modulation and coding scheme) index 7, so at least one AP is reachable for every mobile node.



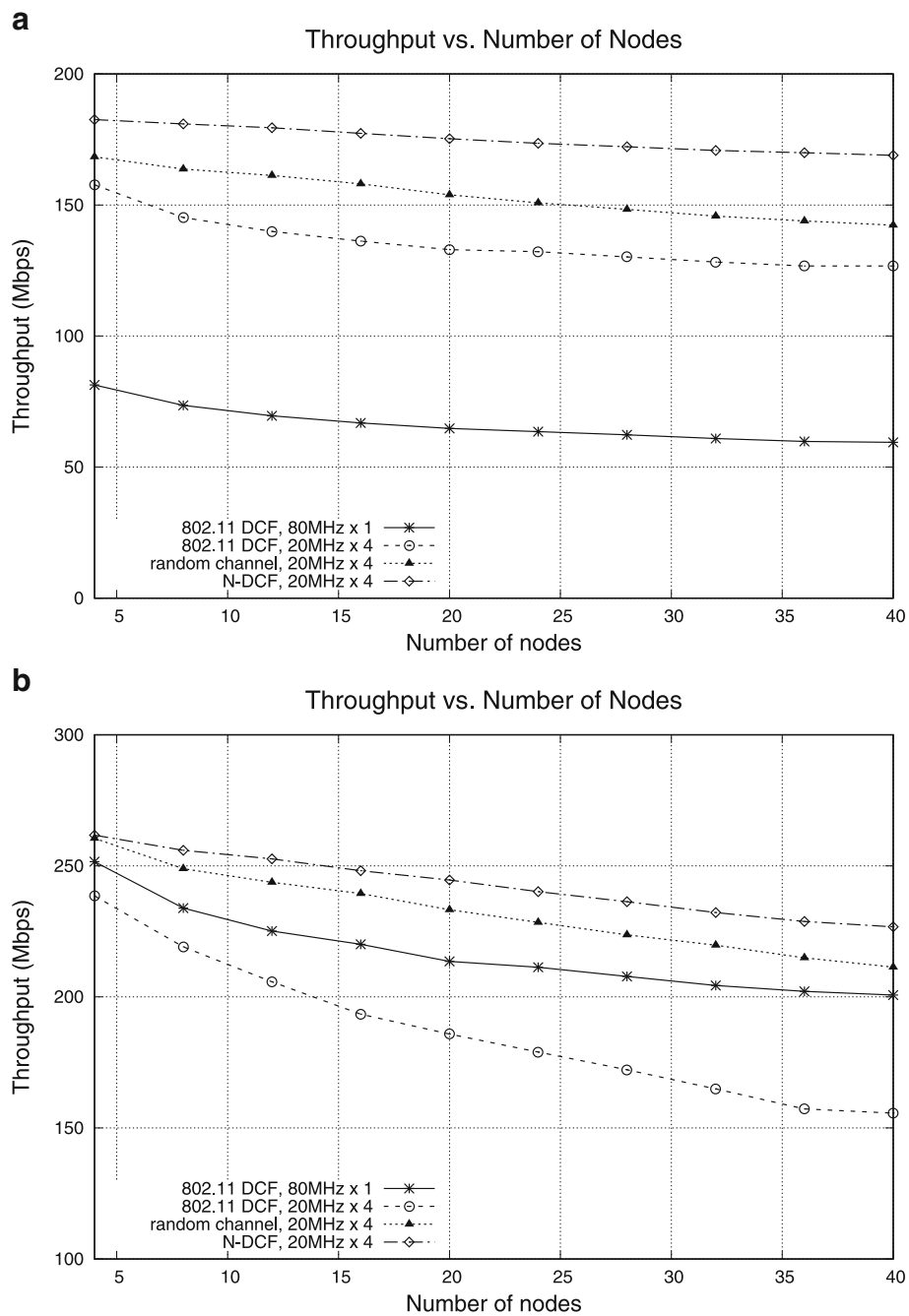
**Fig. 9** The multi-WLAN simulation environment. Four APs are placed in a grid-style locations, and mobile stations are randomly deployed in a 40 m × 40 m area. Transmission range is approximately 30 m for packets using MCS index 7

Every mobile node is always backlogged with UDP traffic to send to its associated AP. MCS was fixed at index 7, and the short guard interval is used. With that configuration, the raw data rate is 72.2 Mbps for 20 MHz, and 325 Mbps for 80 MHz channels. We have compared N-DCF with three other schemes. The first one is the original 802.11 DCF with a single 80 MHz channel. The second one is also the original 802.11 DCF, but we use four 20 MHz channels. Each node evenly distributes packets to all radios, and DCF is operated independently at each radio. The third one is called “random channel” scheme, where we have four 20 MHz channels, and each node randomly selects one channel and only send packets on the selected channel. As discussed before, this scheme can reduce contention at each channel. However, since a node cannot access other channels even if there is no traffic on those channels, resource may be underutilized. In the simulations, all nodes have backlogged traffic, which is the best case for the random channel scheme. Finally, N-DCF uses four 20 MHz channels and performs random channel contention and probabilistic privileged access. The probability of acquiring privilege is set to 50%, unless otherwise specified. For performance metrics, we measure average system throughput, and Jain’s fairness index [7]. The fairness index is calculated using throughput measured for 1 s. Achieving short-term fairness is important, because one can always increase system throughput by having a single node dominate the entire resource. In the graphs showing results, each point is an average of 100 simulation runs with different topology and seed for the random number generator.

**5.2 Results**

Figures 10 and 11 show average throughput and fairness index of protocols when number of nodes is varied. The result is shown for small packets (3000 bytes) and large packets (65,535 bytes). It can be observed from Fig. 10a, b that when the packet size is small, using multiple narrow channels is better than using a single wide channel of the same total bandwidth, because percentage of MAC overhead is smaller in narrow channels. On the other hand, when the packet size is large, using a single wide channel is better, because when using narrow channels, throughput is lost due to guard band. N-DCF achieves higher throughput than 802.11 DCF regardless of packet size. Also, N-DCF achieves higher throughput compared to the random channel scheme where each node only contends for its assigned channel. The additional throughput is gained by reducing channel idle time using privileged access.

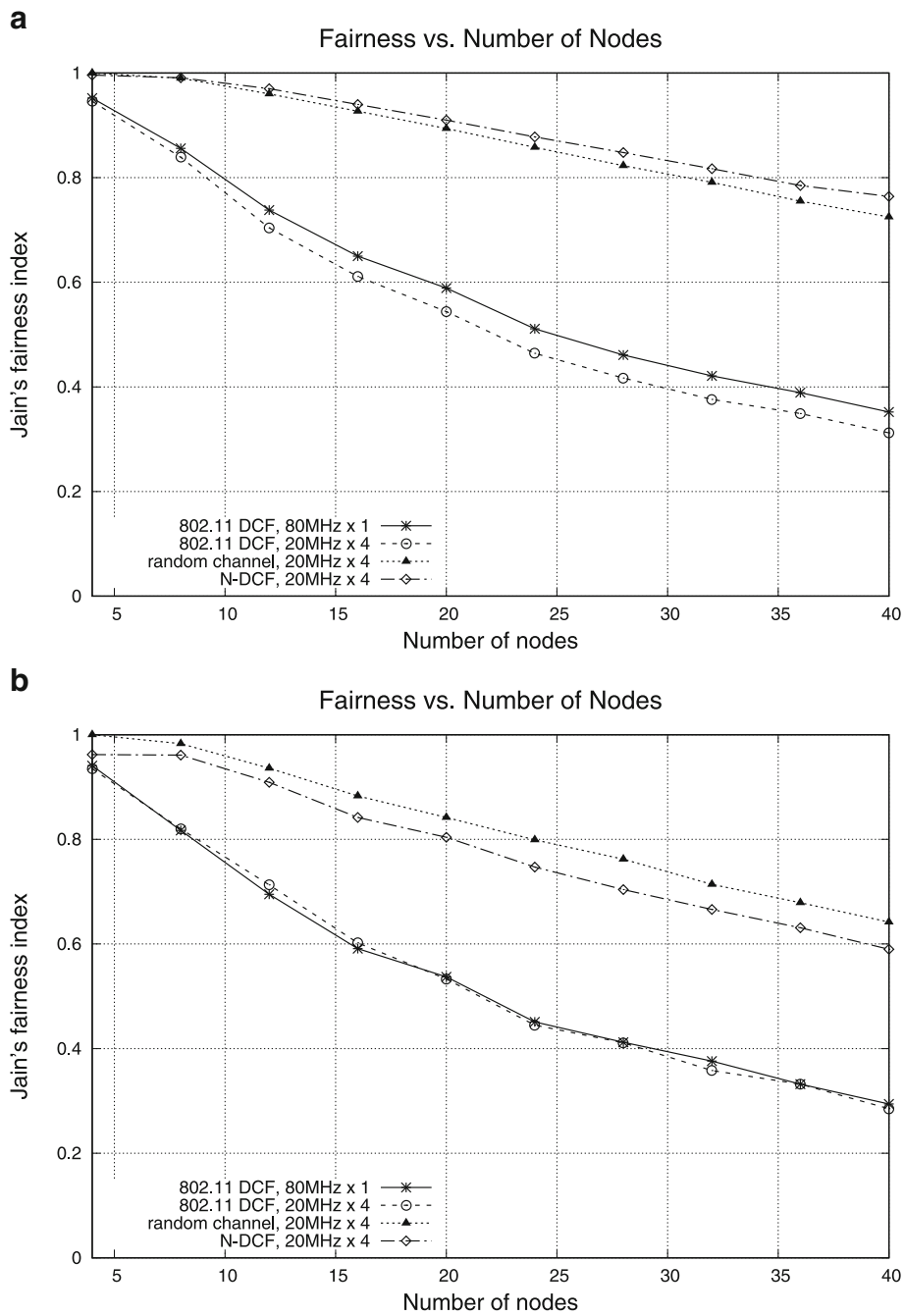
Figure 11a, b shows the short-term fairness of the protocols. As the node density increases, the fairness index for all schemes decrease gradually. Notably, the fairness of 802.11 DCF is much lower than the random channel



**Fig. 10** Average throughput varying number of nodes. **a** A-MPDU size = 3000 bytes. **b** A-MPDU size = 65,535 bytes

scheme and N-DCF. This is due to the behavior of exponential random backoff. When a node is caught in a packet collision, its chance of accessing the channel becomes lower because the contention window is doubled. So by reducing packet collisions, short-term fairness can be improved as well as throughput. Fairness of the random channel scheme and N-DCF are similar, because they both reduce contention by distributing contention overhead across channels.

Figure 12 shows the impact of number of radios on throughput and fairness. Number of nodes is fixed at 20, and the probability of acquiring the privilege is fixed at 50%. Channel width is fixed at 20 MHz, and the results are shown for small and large packets. For the three schemes, throughput and fairness are almost the same with a single channel. As the number of radios increase, the random channel scheme and N-DCF outperform 802.11 DCF. The gap of throughput between schemes is different for small

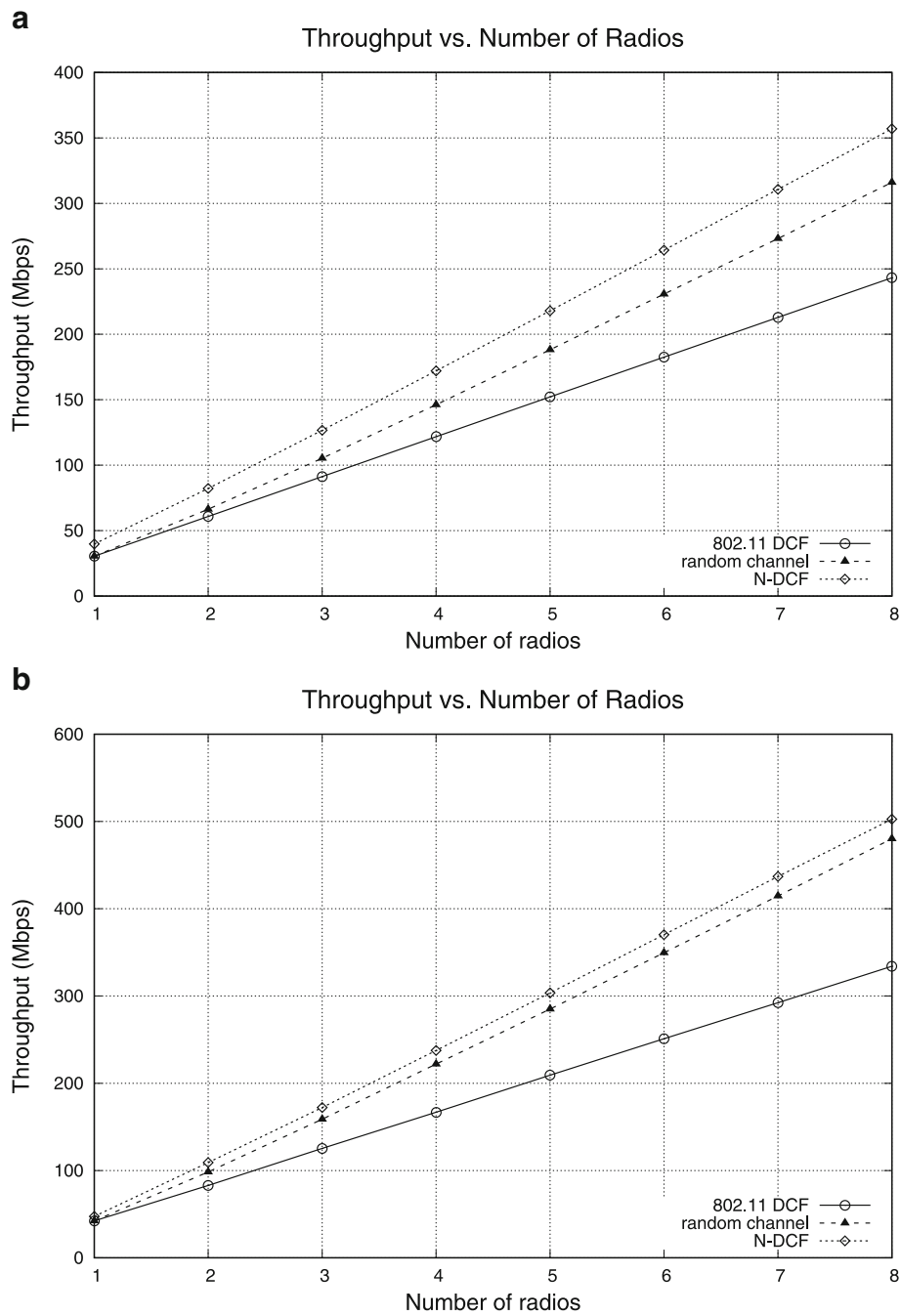


**Fig. 11** Jain's fairness index varying number of nodes. **a** A-MPDU size = 3000 bytes. **b** A-MPDU size = 65,535 bytes

and large packet size. When the packet size is small, both channel idle time and packet collisions significantly affect the system throughput. On the other hand, when the packet size is large, effect of channel idle time becomes minimal and the throughput gain mostly comes from reducing packet collisions. That is why the gap is larger between the random channel scheme and N-DCF when the packet size is small. Figure 13 shows the fairness index

of the protocols. For 802.11 DCF, the fairness index stays the same regardless of number of radios, because the collision rate is the same. For the random channel scheme and N-DCF, the fairness index increases with number of radios up to approximately 95%, meaning the resource is fairly shared among nodes.

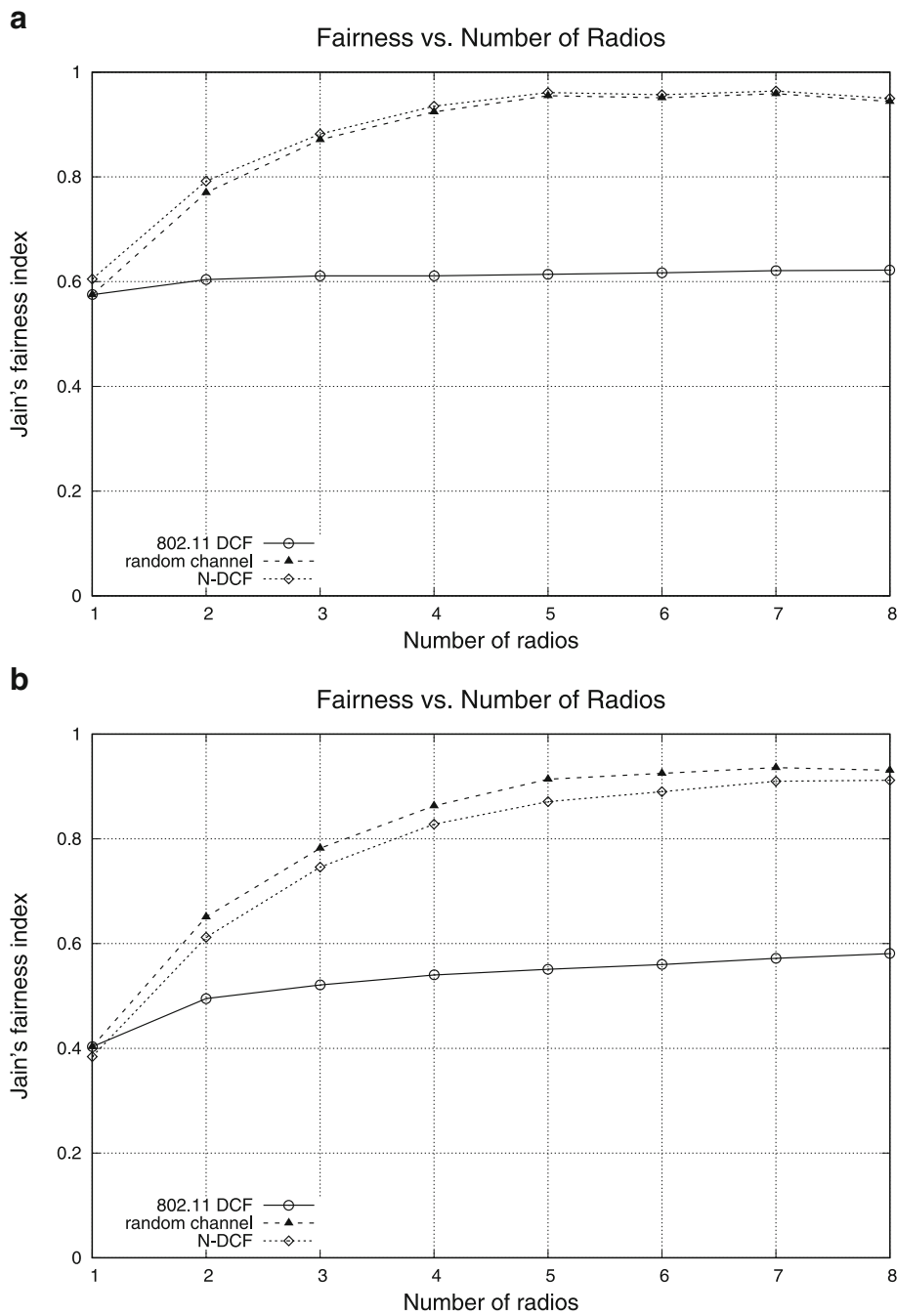
In N-DCF, the privilege of accessing the next channel is gained probabilistically, where the probability  $p$  is a system



**Fig. 12** Average throughput varying number of radios. **a** A-MPDU size = 3000 bytes. **b** A-MPDU size = 65,535 bytes

parameter. Figure 14 shows the impact of this parameter on throughput and fairness. In this experiment, the number of nodes is fixed at 20 and we use four radios with 20 MHz channel width. Figure 14a shows that throughput increases when we increase  $p$ , because more priority is to the privileged nodes. The throughput improvement is achieved at the cost of fairness, because with more privileged accesses, the chance for contending nodes to access

the channel is reduced. Figure 14b shows the fairness when the probability of privileged access is varied. When  $p$  is 100%, Jain's fairness index is 0.05, meaning a single node dominates the entire resource. It is promising to observe that when the probability is reduced from 100%, the fairness quickly increases. So for non-delay-sensitive traffic, we can use high probability for high throughput without severely damaging fairness.



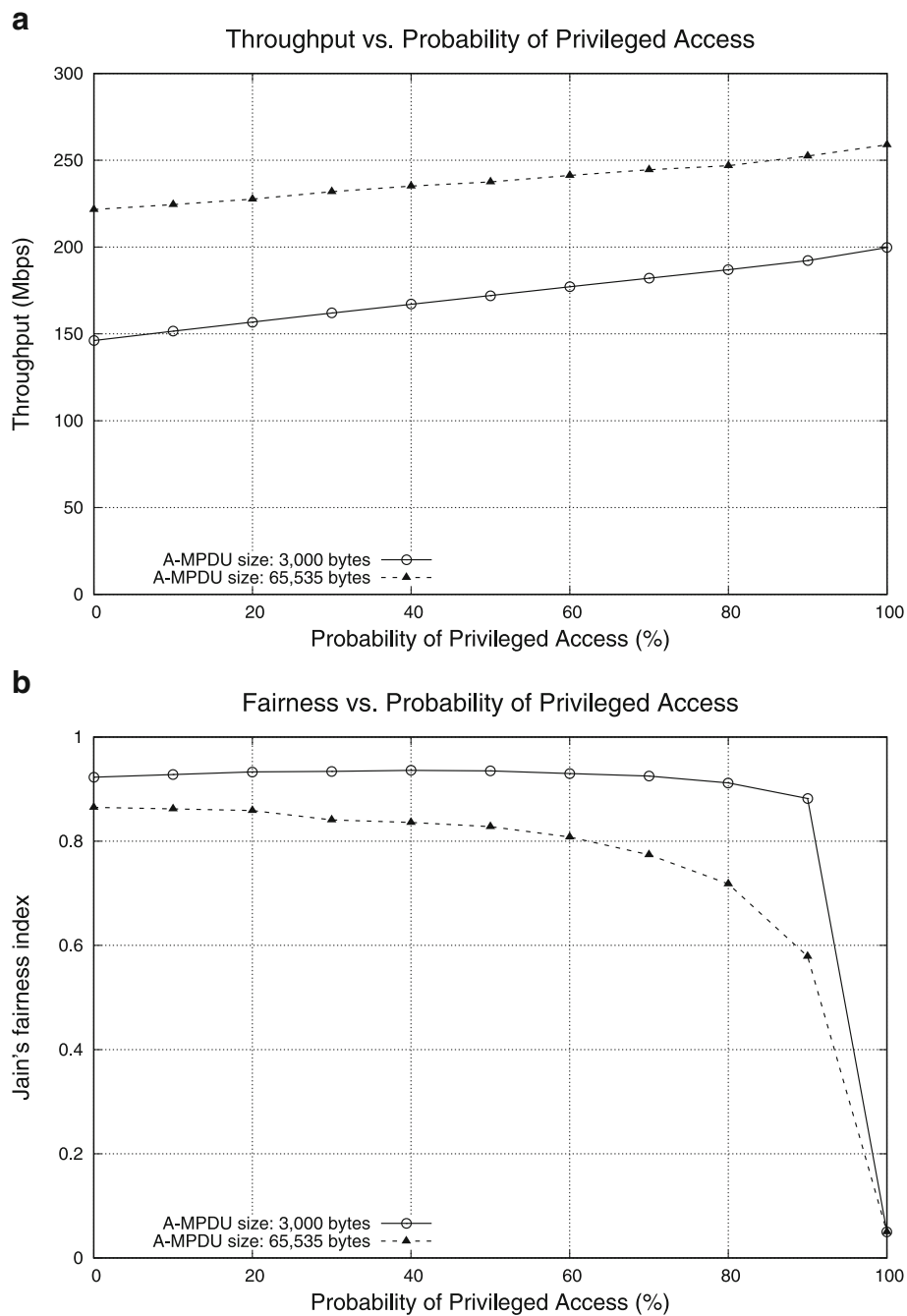
**Fig. 13** Jain's fairness index varying number of radios. **a** A-MPDU size = 3000 bytes. **b** A-MPDU size = 65,535 bytes

### 5.3 Ad hoc networks

All experiments up to now were conducted on a multi-WLAN environment. However, N-DCF also works on ad hoc networks, because the protocol does not require special behavior from access points. In this section, similar experiments were conducted on an ad hoc network described in Fig. 15. Nodes are randomly placed in a 20 m × 20 m area, and half of the nodes send UDP traffic to

other half nodes. The sender-receiver pairs are randomly chosen.

The results shown in Figs. 16 and 17 are similar to the results from multi-WLAN (Figs. 10 and 11). It is shown from the graphs that depending on the packet size, a single wide channel or multiple narrow channels could achieve higher throughput when using 802.11 DCF. However, N-DCF achieves higher throughput than 802.11 DCF



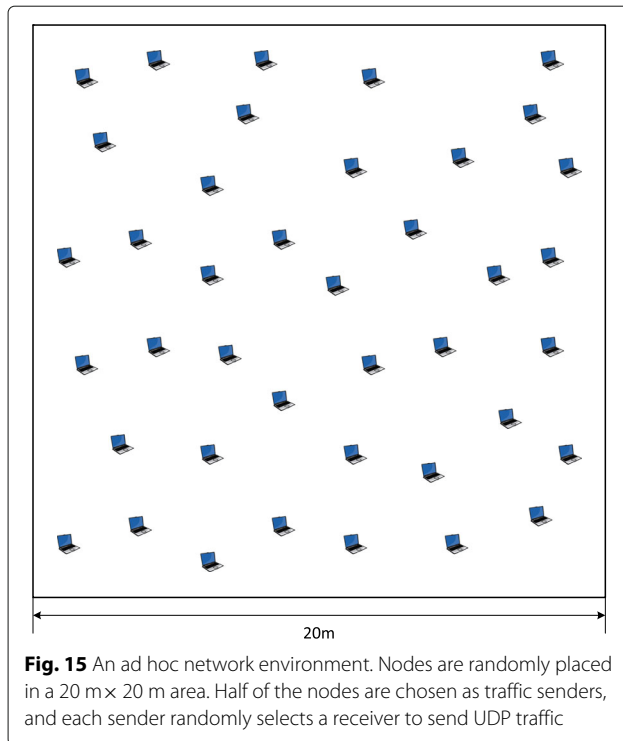
**Fig. 14** Average throughput and Jain's fairness index varying probability of privileged access. **a** Throughput. **b** Fairness

regardless of packet size and also outperforms the random channel scheme. In terms of fairness, N-DCF achieves better fairness compared to 802.11 DCF and is comparable to the random channel scheme. Higher throughput of N-DCF is achieved by reducing packet collision rate and channel idle time. Better fairness comes from reduced packet collision rate, because collisions cause unfairness among nodes as discussed earlier.

#### 5.4 Discussions

The simulation results have shown that N-DCF can achieve better performance compared to the conventional 802.11 DCF in both throughput and fairness. In this section, we discuss several issues that were not covered in the experiments. First, the assumption of N-DCF for privileged access is that channels have similar propagation characteristics. If a node wins a channel and successfully





**Fig. 15** An ad hoc network environment. Nodes are randomly placed in a 20 m × 20 m area. Half of the nodes are chosen as traffic senders, and each sender randomly selects a receiver to send UDP traffic

transmits a packet on the channel, it should be an indication that the node will successfully transmit a packet on the next channel. If the channels are in different frequency band, for example 2.4 GHz and 5 GHz, winning a channel may not lead to successful transmission on the other channel. Also, if nodes use different parameters such as transmit power, carrier sense threshold, and MCS index, it could also lead to failures for privileged access. Thus, when using N-DCF, a node should apply the same parameters to all channels. (It is possible that different nodes use different parameters.)

Second, it is possible that nodes running N-DCF could co-exist with ordinary 802.11 nodes. While the system could still work, 802.11 nodes will have less chance of accessing the channel because they do not gain privilege on the channels. One simple idea to improve fairness is to control the probability of privilege acquisition ( $p$ ) based on the ratio of N-DCF and ordinary 802.11 nodes. A high  $p$  can be used if there are more N-DCF nodes, and a low  $p$  can be used if there are more 802.11 nodes. There will be a trade-off between system throughput and fairness among nodes.

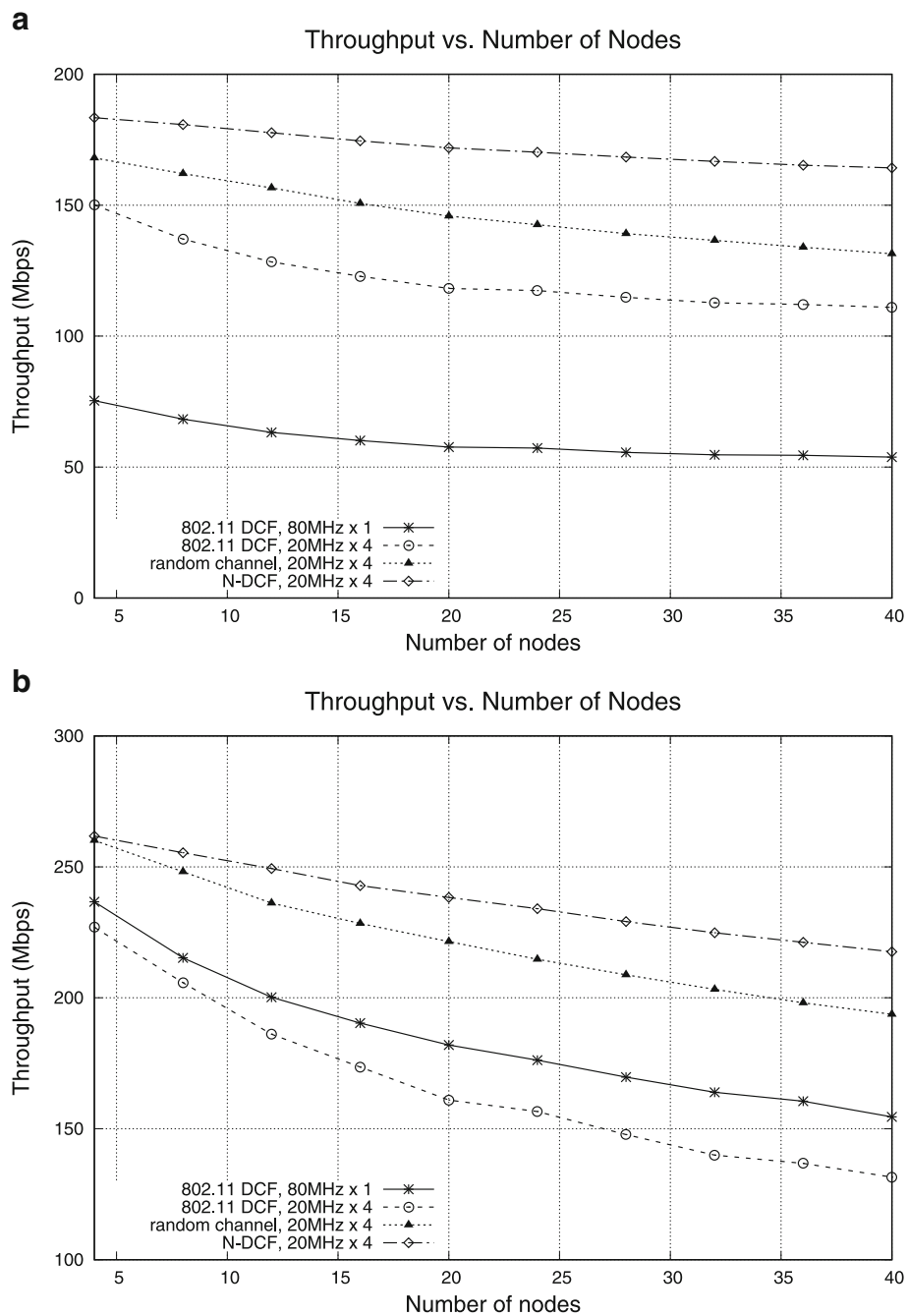
## 6 Related work

It has been 20 years since the first wireless LAN standard was released. Since then, the data rate of wireless LANs has rapidly grown from 1 Mbps to more than 1 Gbps. However, the system spectral efficiency has not grown at the same magnitude as the link spectral efficiency. The

CSMA/CA-based medium access control is still being used, although techniques such as frame aggregation have boosted the system throughput to some extent [8]. As the link rate increases, MAC overhead becomes a serious problem for system performance. To reduce MAC overhead and improve system spectral efficiency, IEEE 802.11 has started a study group called high-efficiency WLAN (HEW) which later became the 802.11ax task group [9].

The idea of using multiple radios has attracted a lot of interest among researchers as an approach to improve system throughput. Since orthogonal channels do not interfere with each other, simultaneous transmissions are possible in the same region. However, because nodes listening on different channels cannot hear each other, protocol support is needed to synchronize channels between senders and receivers. Many different protocols were proposed to address medium access control, routing, and channel assignment in the multi-channel wireless network environments. MMAC [10] and SSCH [11] are medium access protocols that utilize multiple channels using a single network interface. The major challenge in these protocols is to synchronize channels between nodes through channel switching so that they can exchange packets on the same channel. When nodes have multiple interfaces, synchronizing channels between nodes becomes much simpler. For example, one of the interfaces can be fixed on a channel while other interfaces dynamically change channels. The fixed interface can be used to receive packets which removes the need for temporal synchronization of channels [12]. MUP [13] is a protocol that manages multiple network interfaces by assigning channels and scheduling packets across the available channels. In multi-hop wireless networks, routing protocols can consider channel diversity when selecting routes. For example, WCETT [14] considers interference among links that use the same channel, when selecting the routes.

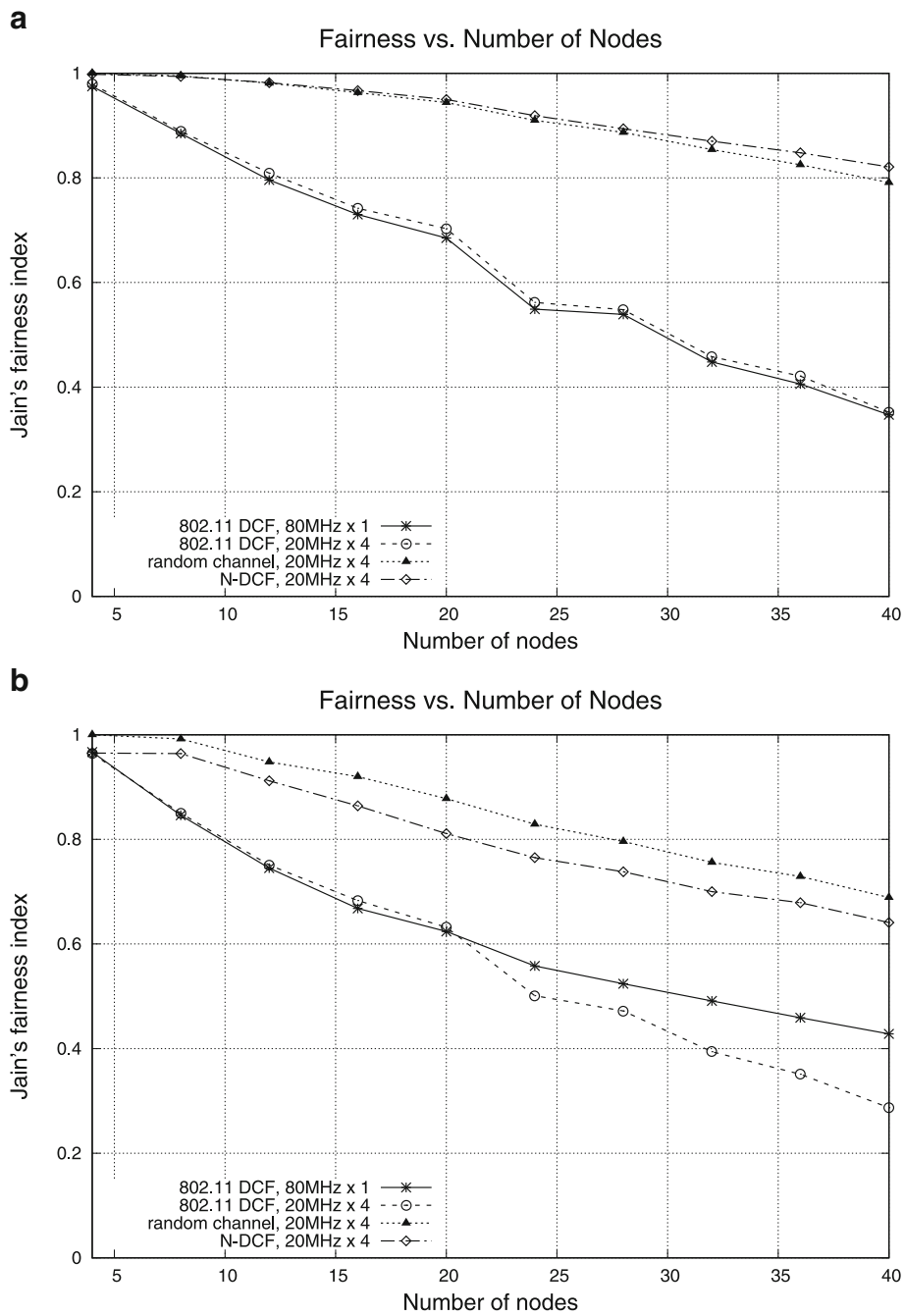
With the advance of software-defined radios (SDR), dynamic spectrum access (DSA) has become another promising approach for improving spectral efficiency. In DSA, channels are frequently switched according to current channel conditions. Not only the center frequency, but also the channel width can be dynamically configured. Moreover, separated frequency bands may be combined to form a single channel. This flexibility of channel selection can significantly improve system throughput, because interference may exist only in a small portion of the channel due to frequency-selective fading [15]. FICA [16] divides a channel into multiple subchannels and lets multiple nodes send packets simultaneously on different subchannels. WiFi-NC [3] goes one step further and builds compound radios where nodes may send and receive independently on narrow sub-channels. FSA [17] allows dynamic channel selection by the sender without previous coordination with the receiver. Spectrum information is



**Fig. 16** Average throughput varying number of nodes for an ad hoc network. **a** A-MPDU size = 3000 bytes. **b** A-MPDU size = 65,535 bytes

contained in the preamble, so that the receiver can identify the spectrum and quickly adapt to it. While the dynamic spectrum access of FSA is limited to the RF band of the receiver, SEER [18] allows outband spectrum adaptation. In SEER, nodes can detect preamble sent by the sender even if the nodes are listening on arbitrary channels. The core idea of SEER is to design a preamble that can be detected from a wide range of frequency bands. Once

a node receives the preamble, it can switch its listening channel based on information carried in the preamble. Ez-channel [19] allocates different sub-channels to potential hidden and exposed terminals, in order mitigate the negative effect of hidden and exposed terminals and therefore improve spectral efficiency. Virtual Duplex [20] divides a channel into two sub-bands and allocates downlink and uplink traffic to each sub-band. The bandwidths of the



**Fig. 17** Jain's fairness index varying number of nodes for an ad hoc network. **a** A-MPDU size = 3000 bytes, **b** A-MPDU size = 65,535 bytes

subchannels are configured in order to overcome traffic asymmetry and maximize spectral efficiency. FSS [21] introduces an algorithm for efficiently allocating spectrum to nodes based on utility maximization framework which considers opportunity from using non-contiguous spectrum chunks and channel wastage due to guardband. DyB [22] allows nodes to start transmission whenever there are some idle narrow channels available. While

transmitting, the node can gradually increase channel width if new narrow channels become available.

Many ideas were proposed which aim to reduce the MAC overhead of CSMA-based wireless LANs, especially regarding the random backoff scheme of 802.11 DCF. WiFi-Nano [23] reduces slot time to 800 ns instead of 9  $\mu$ s in current standards. Since backoff time is proportional to slot time, using short slot time can reduce idle

time and thus improve system throughput. Back2F [24] removes temporal backoff using frequency domain backoff. Instead of waiting for a random number of slots, each node transmits a tone on the selected subcarrier. By listening to the tones, a node can determine if it has selected the smallest number which will win the channel. HiBo [25] uses hierarchical backoff in order to reduce the average backoff time. In HiBo, nodes pick random numbers from a small range which will result in multiple nodes selecting the same number. The nodes who selected the same number move to the next round and perform another contention. HiBo decreases both idle time and collision rate, thereby improving the system throughput.

Existing protocols do not utilize collaboration between multiple radios to reduce MAC overhead. Protocols using multiple radios or dynamic spectrum access mostly focus on assigning the best channel to links. Also, protocols that reduce MAC overhead are mostly single-channel-based schemes. Different from existing protocols, N-DCF utilizes collaboration between multiple radios to reduce MAC overhead and improve system spectral efficiency.

## 7 Conclusions

For high-speed wireless networks, MAC overhead such as channel idle time and packet collisions becomes a serious factor degrading the system performance. Frame aggregation techniques included in recent WLAN standards can reduce channel idle time, but the negative effect of packet collisions remains and significantly affects system spectral efficiency. Using multiple narrow channels has similar benefits with frame aggregation, which reduces the percentage of time spent on random backoff and interframe spacing. However, just using multiple channels do not reduce packet collisions, because all nodes may contend on all channels. In the proposed protocol N-DCF, each node randomly chooses one of the channels as its contention channel. This scheme distributes nodes across channels thereby reducing the collision rate. However, resource may not be fully utilized because a node cannot access other channels even if they are idle. Thus, N-DCF introduces privileged access, in which a node acquires immediate access to the next channel once it successfully finishes transmission on a particular channel. The privilege is obtained probabilistically, in order to give chance to non-privileged nodes who are contending for the channels. Regardless of packet size, N-DCF outperforms original 802.11 DCF, as well as the “random channel” scheme where each node randomly chooses a single channel for communication. For future work, we are going to investigate the use of narrower channels, such as 1 MHz channels. In the performance evaluations, we only used four channels. However, N-DCF can gain more benefit when there are more channels. Using very narrow

channels can be helpful, but it comes with a cost such as longer preamble time, longer time slots and more guard bands. We are going to study this trade-off to see if there is an optimal channel configuration which achieves the maximum throughput. There are many techniques that can potentially further improve performance of N-DCF. For example, choosing contention channel intelligently instead of randomly can potentially mitigate the effect of hidden and exposed terminals. Also, probability of privileged access can be controlled according to traffic requirements. Different probability can be assigned to different nodes to enable prioritized service.

In terms of implementation, N-DCF cannot be directly implemented onto current off-the-shelf WLAN devices mainly because the MAC is implemented as hardware in current devices. To run N-DCF, MAC modules of the interfaces should support message exchange in order to signal privilege information. Also, multiple interfaces should share a single packet queue. These features are simple to implement, but they should be included in the standard in order to be implemented on commercial devices. In the near future, we are going to implement N-DCF on a testbed based on software-defined radio as a proof-of-concept and evaluate its performance on a real-world environment.

## Abbreviations

MAC: Medium access control; LAN: Local area network; DCF: Distributed coordination function; DIFS: DCF inter-frame spacing; SIFS: Short inter-frame spacing; MCS: Modulation and coding scheme; SGI: Short guard interval; ACK: Acknowledgment; BA: Block ACK; CSMA: Carrier sense multiple access; CA: Collision avoidance; HEW: High-efficiency WLAN; SDR: Software-defined radio; DSA: Dynamic spectrum access

## Funding

This work was supported in part by the National Research Foundation of Korea under grant NRF-2016R1D1A1 B03930289 and Hallym University Research Fund (HRF-206010-009).

## Availability of data and materials

The simulation code can be found at <https://github.com/jungminso/ns-3-narrow>.

## Authors' contributions

JS designed the protocol and implemented modules in the ns-3 for performance evaluation. AE helped running the simulations. Both authors read and approved the final manuscript.

## Authors' information

Jungmin So is a faculty member in the Department of Computer Engineering at Hallym University, Korea. Ayinebyona Eliab is a Ph.D candidate in the Department of Computer Engineering at Hallym University, Korea.

## Competing interests

The authors declare that they have no competing interests.

## Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Received: 21 November 2017 Accepted: 30 August 2018

Published online: 19 October 2018

## References

1. IEEE 802.11 Working Group, Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, Amendment 4: Enhancements for Very High Throughput for Operations in Bands below 6GHz, IEEE Standard 802.11 (2013)
2. G. Hosseinabadi, N. Vaidya, in *Proc. of the 5th International Conference on Communication Systems and Networks (COMSNETS)*. Token-DCF: An opportunistic MAC protocol for wireless networks (Bangalore, 2013), pp. 1–9
3. K. Chintalapudi, B. Radunovic, V. Balan, M. Buettener, S. Yerramalli, V. Navda, R. Ramjee, in *Proc. of the 9th USENIX conference on Networked Systems Design and Implementation (NSDI)*. WiFi-NC: WiFi over narrow channels, (San Jose, 2012), p. 4
4. The ns-3 simulator. Available: <https://www.nsnam.org>. Accessed 20 Nov 2017
5. V. Raman, N. Vaidya, in *Proc. of the 8th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON)*. WiSP: A protocol for overcoming MAC overheads using packet size dependent channel widths (Salt Lake City, 2011)
6. J. So, J. Park, A. Eliab, Wireless interface bonding supporting in-order delivery and automatic load balancing. *IEEE Trans. Veh. Technol.* **66**(6), 5190–5203 (2017)
7. R. Jain, D. Chiu, W. Hawe, A quantitative measure of fairness and discrimination for resource allocation in shared computer systems, Digital Equipment Corporation Research Report TR-301 (1984)
8. D. Skordoulis, Q. Ni, H. hwa Chen, A. P. Stephens, C. Liu, A. Jamalipour, IEEE 802.11n MAC frame aggregation mechanism for next-generation high-throughput WLANs. *IEEE Wirel. Commun.* **15**(1), 40–47 (2008)
9. B. Bellalta, IEEE 802.11ax: high-efficiency WLANs. *IEEE Wirel. Commun.* **23**(1), 38–46 (2016)
10. J. So, N. Vaidya, in *Proc. of the 5th ACM international symposium on Mobile ad hoc networking and computing (MobiHoc)*. Multi-channel MAC for ad hoc networks: handling multi-channel hidden terminals using a single transceiver, (Tokyo, 2004), pp. 222–233
11. P. Bahl, R. Chandra, J. Dunagan, in *Proc. of the 10th annual international conference on Mobile computing and networking (MobiCom)*. SSSH: slotted seeded channel hopping for capacity improvement in IEEE 802.11 ad-hoc wireless networks, (Philadelphia, 2004), pp. 216–230
12. P. Kysanur, J. So, C. Chereddi, N. H. Vaidya, Multichannel mesh networks: challenges and protocols. *IEEE Wirel. Commun.* **13**(2), 30–36 (2006)
13. A. Adya, P. Bahl, J. Padhye, A. Wolman, L. Zhou, in *Proc. of the 1st international conference on broadband networks (BroadNets)*. A multi-radio unification protocol for IEEE 802.11 wireless networks, (San Jose, 2004)
14. R. Draves, J. Padhye, B. Zill, in *Proc. of the 10th annual international conference on Mobile computing and networking (MobiCom)*. Routing in multi-radio, multi-hop wireless mesh networks, (Philadelphia, 2004), pp. 114–128
15. E. Chai, J. Lee, S. Lee, R. Etkin, K. Shin, in *Proc. of the 18th annual international conference on Mobile computing and networking (MobiCom)*. Building efficient spectrum-agile devices for dummies, (Istanbul, 2012)
16. K. Tan, J. Fang, Y. Zhang, S. Chen, L. Shi, J. Zhang, Y. Zhang, in *Proc. of the ACM SIGCOMM conference*. Fine-grained channel access in wireless LAN, (New Delhi, 2010), pp. 147–158
17. S. Yun, D. Kim, L. Qiu, in *Proc. of the 19th annual international conference on Mobile computing and networking (MobiCom)*. Fine-grained spectrum adaptation in wifi networks, (Miami, 2013), pp. 327–338
18. W. Wang, Y. Chen, Z. Wang, J. Zhang, K. Wu, Q. Zhang, Wideband spectrum adaptation without coordination. *IEEE Trans. Mob. Comput. (TMC)*. **16**(1), 243–256 (2017)
19. S. Fayaz, F. Zarinni, S. Das, Ez-Channel: A distributed MAC protocol for efficient channelization in wireless networks. *Ad Hoc Netw.* **31**(C), 34–44 (2015)
20. A. Flores, E. Knightly, *Virtual duplex: scaling dense WLANs and eliminating contention asymmetry*, (Raleigh, 2014)
21. S. Han, X. Zhang, K. Shin, Fair and efficient coexistence of heterogeneous channel widths in next generation wireless LANs. *Trans. Mob. Comput. (TMC)*. **15**(11), 2749–2761 (2016)
22. P. Huang, X. Yang, L. Xiao, Dynamic channel bonding: enabling flexible spectrum aggregation. *IEEE Trans. Mob. Comput. (TMC)*. **15**(2), 3042–3056 (2016)
23. E. Magistretti, K. Chintalapudi, B. Radunovic, R. Ramjee, in *ACM MobiCom*. WiFi-Nano: reclaiming WiFi efficiency through 800 ns slots, (Las Vegas, 2011)
24. S. Sen, R. Choudhury, S. Nelakuditi, in *Proc. of the 17th annual international conference on Mobile computing and networking*. No time to countdown: migrating backoff to the frequency domain, (Las Vegas, 2011)
25. M. Gowda, N. Roy, R. Choudhury, S. Nelakuditi, in *Proc. of IEEE 24th international conference on network protocols (ICNP)*. Compressing backoff in CSMA networks, (Singapore, 2016)

Submit your manuscript to a SpringerOpen<sup>®</sup> journal and benefit from:

- Convenient online submission
- Rigorous peer review
- Open access: articles freely available online
- High visibility within the field
- Retaining the copyright to your article

---

Submit your next manuscript at ► [springeropen.com](https://www.springeropen.com)

---