DSCR: A More Stable MAC Protocol for Wireless Networks

Xue Yang and Nitin H. Vaidya
Electrical and Computer Engineering Department, and
Coordinated Science Laboratory
University of Illinois at Urbana-Champaign

Technical Report
August 2002

Abstract— **Channel capacity is a scarce resource in wireless networks. However, the way in which IEEE 802.11 DCF uses random backoff time to resolve the channel contention leads to inefficient utilization of this scarce resource. In particular, in a highly loaded network, the portion of channel bandwidth wasted due to collisions is significantly high. Prior work proposes various ways to improve the performance of 802.11. In this paper, we present a different mechanism which uses two "virtual" stages of contention resolution to achieve better and more stable performance than IEEE 802.11 DCF. The proposed mechanism is compatible with 802.11, and can be incorporated in IEEE 802.11 without major changes. Simulation results, as well as some analysis, are presented to demonstrate the effectiveness of this mechanism.**

I. Introduction

The fraction of channel bandwidth used by successful packet transmissions is known as the utilization of the channel [1]. Since wireless networks typically have relatively low channel capacity, improving channel utilization is important. This paper presents a MAC protocol that improves on IEEE 802.11.

IEEE 802.11 Wireless LAN standard [2] defines a distributed random access MAC protocol: DCF (Distributed Coordination Function). DCF uses *binary exponential backoff* (BEB) algorithm to resolve channel contention. In DCF, after the channel has been idle for DIFS (DCF Interframe Space) duration, a station wanting to access the channel generates a random backoff counter. This backoff counter corresponds to the number of *slots*[1] this station has to wait before its transmission. The backoff counter is uniformly distributed over the interval [0, CW], where CW represents the *contention window* with initial value CW_{min}. When the channel is idle, the backoff counter is decremented by 1 after each slot. The backoff counter is frozen when the channel becomes busy, and is decremented when the channel is idle again for DIFS duration. Once the backoff counter reaches zero, the station is allowed to transmit. If multiple stations happen to start their transmissions in the same slot, a collision may occur. In the case of a collision, the colliding packet may be retransmitted. In addition, the contention window, CW, is exponentially increased by a factor of 2, until it reaches the maximum value denoted by CW_{max}. Once a packet is successfully transmitted by a station, CW at that station is reset to CW_{min}.

Some prior research work analyzes the performance of IEEE 802.11 DCF [3], [4], [5], [6], [7], [8], [9], [10]. Clearly, the choice of contention window is critical to the performance of 802.11. When there are few competing stations in the network, a smaller CW will reduce the idle channel time and enable better usage of channel bandwidth. When the number of competing stations increases, a larger CW is preferred to reduce the collision probability. An appropriate choice of CW can optimize the performance of 802.11. However, the optimum value of CW changes with the network size and 802.11 operates far from this optimum point [11]. In particular, in a highly loaded network, the collision probability increases significantly, degrading the performance of 802.11. This can be confirmed from the simulation results presented later in this paper, and other papers as well [6], [3], [11].

This paper proposes a dual stage contention resolution algorithm, which includes two "virtual" contention resolution stages. The first stage (stage 1) helps to reduce the number of stations entering the second stage (stage 2). Only the stations in stage 2 may try to transmit a packet. The number of stations in the second stage is kept relatively small compared to the total network size. Therefore, the MAC protocol can exhibit higher stability in the sense that the throughput is less sensitive to the network size.

This research is supported in part by National Science Foundation grant 01-96410

[1] A slot is a fixed duration of time defined in IEEE 802.11

The reason we use the word "virtual" here is because the two stages overlap in time. At a certain time, some stations will be in stage 2 while others are in stage 1. Only those stations in stage 2 will contend for channel access. We use both simulation and analysis results to show that the number of stations contending for channel in the second stage will be less than 26 when the total number of contending stations in the network is less than 256.

A new MAC protocol, named DSCR (Dual Stage Contention Resolution) is presented in this paper. Compared to 802.11 DCF, simulation results show that DSCR achieves better and more stable performance in terms of channel utilization for networks containing up to 256 contending stations. Unlike HIPERLAN/1 [12], which is also a MAC protocol with two contention resolution stages, DSCR does not rely on blackburst mechanism. The simulation results show that DSCR remains better performance even in the presence of "hidden terminals", which distinguishes it from HIPERLAN/1.

The rest of the paper is organized as follows. Relevant terms are defined in Section II. We give an introduction to IEEE 802.11 DCF in Section III. The related work is discussed in Section IV. Section V describes DSCR MAC protocol in detail. In Section VI, we use simulation results to demonstrate the effectiveness of DSCR. We also analyze the behavior of the first stage of DSCR in Section VII. In Section VIII, DSCR-F, which is a modified version of DSCR to improve fairness, is described. We simulated both DSCR-F and IEEE 802.11 DCF in random topologies, and the results are presented in section IX. Finally, conclusion is presented in Section X.

II. Relevant Terms

A *transmission cycle* is defined as the time period during which stations compete for channel access and finally one packet is successfully transmitted. A transmission cycle is composed of the time periods during which channel is idle, has collisions, and the time period spent for a packet's successful transmission. As an example, the transmission cycle in Figure 1 consists of two collisions, the successful transmission and the idle time between consecutive transmission attempts. Once a packet has been transmitted successfully, a new "transmission cycle" begins.

Throughout this paper, we use the term "active station" to represent a station that has backlogged packets waiting to be transmitted.

III. Overview of IEEE 802.11 DCF

In this section, we briefly describe the related features of IEEE 802.11 DCF. For more details, please refer to [2].

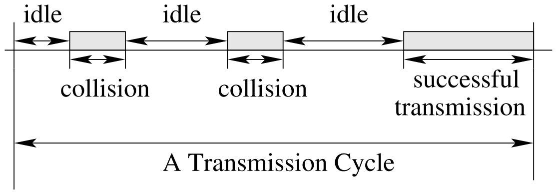


Fig. 1. A Transmission Cycle

IEEE 802.11 DCF defines two access methods: *basic access method* and *RTS/CTS access method*. The basic access method involves only Data/ACK exchange, in which data packets are transmitted when channel access has been obtained. ACK frames follow successful data packet receptions. In the RTS/CTS access method, RTS (Request To Send) and CTS (Clear To Send) frames are exchanged before Data/ACK packets. RTS and CTS frames contain a duration field that defines the period of time when the medium is to be reserved to transmit the actual Data frame and the returning ACK frame. Stations which overhear RTS/CTS frames defer transmission for this period. This mechanism is referred to as the "virtual carrier sense" and it is implemented using "Network Allocation Vector" (NAV). The duration field is also available in the MAC header of Data and ACK frames. A station updates the NAV with the duration field specified in the overheard frames.

The carrier sense mechanism in IEEE 802.11 includes physical carrier sense and virtual carrier sense. After the channel is sensed to be idle for a DIFS (DCF Interframe Space) duration, the backoff procedure is invoked for each backlogged station. As we explained in Section I, 802.11 DCF uses *binary exponential backoff* algorithm to resolve channel contention. A shorter interframe space, SIFS, is used to separate transmissions belonging to a single dialog (e.g., CTS, Data and ACK frames in the case of RTS/CTS access method). Figure 2 illustrates the RTS/CTS access method of IEEE 802.11 DCF.

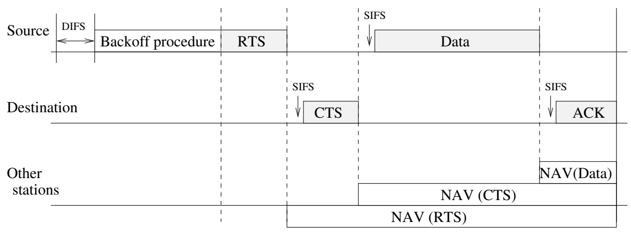


Fig. 2. RTS/CTS Access Method of IEEE 802.11 DCF

IV. Related Work

To achieve a better channel utilization, the time for which the channel is wasted in idle or collision states needs to be reduced. However, there is a tradeoff involved. If stations transmit aggressively, channel idle time will be small but collision time will be large, and vice versa. In [10], it is argued that the network can operate at a point close to the optimum if the time wasted on idle periods is equal to the time spent on collisions. Based on this argument, [10] proposes a scheme in which each station continuously observes the channel activity to estimate the average idle/collision period length. Then at the end of every transmission cycle, each station computes its current estimation for the number of active stations (N) using the analytical results in [11]. Recall that the contention window (CW) in 802.11 DCF is critical to its performance and closely related to N. An optimum CW can be chosen based on the correct estimation of N. As a result, the channel utilization of 802.11 DCF can be optimized. However, the scheme in [10] requires every station to continuously observe channel activities. Moreover, the existence of hidden terminals makes channel sensing unreliable. An earlier work [9] also proposes a similar scheme.

The BEB algorithm of 802.11 DCF adjusts contention window rapidly. When a collision occurs, each colliding station doubles CW. When a station successfully transmits a packet, its CW is reset to a minimum value (CW_{min}). This CW adjustment procedure causes a large variation of contention window. Especially in a highly loaded network, a station has to go through several collisions repeatedly to reach the appropriate contention window, which leads to much wasted channel bandwidth in collision state. MACAW [13] suggests a gentler adjustment algorithm to avoid such large oscillations of contention window. Specifically, CW is exponentially increased by a multiplicative factor of 1.5 upon a collision and linearly decreased by 1 after each successful transmission. MACAW also suggests a "contention window copy" scheme so that each station shares the same CW value. To simulate MACAW, we modify the CW adjustment algorithm in ns-2 simulator [14] and add a CW field in the Data/ACK frame. A station sets it own CW using the value from the overheard Data/ACK frames. The simulation results for 802.11 and MACAW are compared in Figure 3. In the simulation, we use the RTS/CTS access method. Channel bandwidth is set to 11 Mbps and payload packet size is 512 bytes. As specified in 802.11 standard [2], CW_{min} is 31 and CW_{max} is 1023. According to [13], MACAW should use a small value for CW_{min}, which is set to 3 in our simulation. CW_{max} of MACAW is set to 511 since we simulate up to 256 active stations. The number of active stations (denoted as N) is increased from 1 to 256 (all stations are at the same physical location). We can see that 802.11 achieves its maximum aggregate throughput when N is 4. When N increases, its performance degrades. When N is 256, the aggregate throughput of 802.11 drops to 62% of its peak point.

On the other hand, MACAW has better performance for large networks with N greater than 128. In such highly loaded networks, MACAW benefits from the gentler adjustment of CW and it can operate with a relatively good choice of contention window size. When N is between 4 and 64, MACAW performs worse than 802.11. Intuitively, while MACAW avoids the repeated collision periods for highly loaded networks, the slow linear decrement of the contention window usually leads to an unnecessarily large CW when network size is moderate. For example, suppose we have a station which initially has an optimum contention window value. After an unlucky collision, this station will exponentially increase CW, and it will take a long time for this station to reduce its CW back to the optimum point using linear decrement. Thus channel bandwidth is unnecessarily wasted in the idle state. The higher throughput of MACAW when N is 1 or 2 is due to its small value of CW_{min}.

MACAW is more stable than 802.11. The performance of 802.11 drops for larger values of N. If we use the same small $CW_{min} = 3$ for 802.11 BEB algorithm, its throughput will drop even faster (compared to the case of $CW_{min} = 31$) when N increases, which is the result that [13] points out. The curve for 802.11 when $CW_{min} = 3$ is also plotted in Figure 3 for reference.

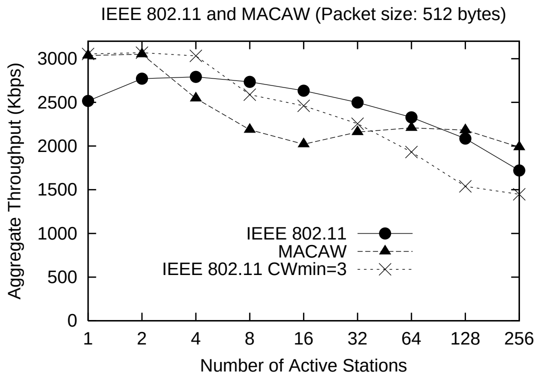


Fig. 3. 802.11 BEB backoff vs. MACAW Mild Backoff (horizontal axis is plotted in log-scale)

As an aggressive backoff algorithm, 802.11 BEB wastes channel bandwidth on collisions in highly loaded networks. On the other hand, a mild backoff algorithm such as MACAW wastes bandwidth in the idle state for a moderate size network. Neither works well for the entire range. This motivates the following question: How to design a backoff algorithm that can adapt to a wide range of network sizes? In this paper, we attempt to answer this question.

A two stage contention resolution scheme may be a reasonable approach to address this question. The first stage helps to reduce the number of stations entering the second stage. Only the stations in stage 2 may try to transmit a packet in each transmission cycle. If only a small fraction out of all active stations attempt to transmit in the second stage, it will be easier to efficiently resolve contention among this small number of stations. Compared to the potentially large variation in the entire network size, if the number of active stations in the second stage remains within a narrow range, then it will be much easier for the MAC protocol to achieve stability.

HIPERLAN/1 [12] is a MAC protocol using two stage contention resolution. It adopted an "Elimination-Yield Non-preemptive Priority Multiple Access"(EY-NPMA) to resolve channel contention. The contention phase consists of two stages: the "elimination" stage and the "yield" stage. In the elimination stage of HIPERLAN/1, a contending station transmits black bursts[2] for a random duration and then listens to the channel in the *elimination survival verification interval*. A contending station survives the elimination stage if and only if the channel is sensed idle in its *elimination survival verification interval*; otherwise, this station is eliminated and withdraws from the competition for the right of transmission in the current channel access cycle. [15] and [16] present detailed analysis and prove that at the end of the elimination stage, only small number (≥ 1) of stations will survive. The yield stage then further resolves contention and reduces the number of stations allowed to transmit to 1 with high probability. [15] shows that HIPERLAN/1 has the desired stability for up to 200 competing stations.

However, the fact that HIPERLAN/1 relies on black burst sensing to implement the elimination stage makes it vulnerable to the "hidden terminal" phenomenon. A pair of hidden stations cannot correctly sense the black bursts sent by each other. As a result, a station may incorrectly regard itself as the survivor of the elimination stage. Compared to an ideal situation without hidden terminals, more stations will enter into the yield stage and subsequent channel competition will be more severe, resulting in degraded performance. [17] shows, using simulation results, the performance of HIPERLAN/1 drops significantly in the presence of "hidden terminals".

[18] has proposed a Fast Collision Resolution (FCR) algorithm. The difference between FCR and 802.11 DCF is that whenever a new busy period is detected (could be either a collision or a packet transmission), *all deferring stations* will exponentially increase their own contention window and generate a new backoff counter. In this way, all deferring stations have larger contention window compared to the winning station. The winning station has higher probability to win channel access again, and naturally the collision probability is low. FCR defines a maximum transmission limit. If a station consecutively succeeds in channel access over this transmission limit, it will change its contention window to maximum value in order to give other stations opportunities to transmit. To save the channel idle time caused by the large value of contention window, FCR exponentially decreases backoff counter after each idle slot when $2CW_{min} - 1$ consecutive idle slots are detected. FCR reduces the collision probability by giving higher transmission probability to any one station during a certain period of time.

The DSCR algorithm proposed in this paper has two contention resolution stages. While only stations in the second stage will contend for the channel access, all stations in the first stage will take turns to enter the second stage. DSCR reduces the collision probability in such a way that only small number of stations contend for the channel in the second stage during each transmission cycle.

V. Dual Stage Contention Resolution MAC Protocol (DSCR)

There are two contention resolution stages in DSCR. When a transmission cycle begins, some stations enter stage 2, while others stay in stage 1. Only the stations in stage 2 may try to transmit a packet.

Ideally, stations only enter stage 2 at the *beginning* of a transmission cycle. After that, stations in stage 1 will stay in stage 1 until next transmission cycle begins. However, it is possible that no station enters the second stage at the beginning of a transmission cycle. In addition, the stations in stage 1 do not know whether there are stations in the second stage or not, unless we have blackbursts or some equivalent mechanisms to indicate the status of each stage (as in HIPERLAN/1). We wish to avoid mechanisms such as blackburst because of the drawback mentioned in Section IV. Therefore, in DSCR, it is still possible for stations in stage 1 to enter stage 2 *during* a transmission cycle, this is what we call "interference" between the two stages in DSCR. DSCR successfully reduces the impact of "interference" on performance, which we will show later.

Without the aid of mechanisms such as blackburst, DSCR implements both the first stage and the second stage

[2]Black bursts are pulses of energy.

using the backoff mechanism. DSCR differs from 802.11 DCF only in the contention resolution procedure. The other functions remain the same.

A. Contention Resolution Stage 1

A backoff counter, named bc_1, is associated with the first contention resolution stage. bc_1 is chosen to be uniformly distributed over the interval [0, CW1]. CW1 is the contention window for stage 1, and has a minimum value $CW1_{min}$ and maximum value $CW1_{max}$. The initial value of CW1 is $CW1_{min}$.

The backoff counter bc_1 is counted down in two ways:

1. Recall that the beginning of a transmission cycle is marked by the end of a successful packet transmission, which a station could determine using its NAV. At the *beginning* of a transmission cycle, each active station reduces bc_1 by K (K is a parameter of DSCR), except for the station which just finished transmitting a packet. Since a station will reset CW1 to $CW1_{min}$ after a successful transmission, if it immediately reduces bc_1 ($bc_1 \in [0, CW1]$) by K, this station will have more chances than other stations to enter stage 2 (i.e., causing unfairness). Later in this paper, we will discuss the impact of the chosen value of K.

2. *During* a transmission cycle, each active station reduces its bc_1 by 1 after each *idle* slot.

Whenever a station's bc_1 becomes less than or equal to 0, this station enters stage 2.

If a station's bc_1 is counted down to zero *during* a transmission cycle (i.e., by above the second method for decrementing bc_1 by 1), we say this station "interferes" with the second stage, as explained at the beginning of this section. Our simulation results show that, with a suitable choice of K, interference only happens to a small fraction of stations entering stage 2. Consequently, it has no major impact on the performance of DSCR. We will present related simulation results in Section VI.

Now assume that there are M stations that enter stage 2 in a transmission cycle. As seen later, among the M stations, one station will eventually win channel access. The remaining $M-1$ stations in stage 2 will return back to the *first* stage, double CW1 and regenerate bc_1. The winning station will finish its packet transmission, then return back to the *first* stage, reset CW1 to $CW1_{min}$ and regenerate bc_1.

Those stations which do not enter the second stage will not participate in channel contention, and hence, will not have their CW1 value changed.

Notice that if there are too many stations entering stage 2 in a transmission cycle, all except the one winning station will double CW1 after the second stage (i.e., all "losers" will double CW1). As a result, fewer stations will enter stage 2 next time.

We will illustrate stage 1 later with an example.

B. Contention Resolution Stage 2

A backoff counter, named bc_2, is associated with the second contention resolution stage.

If a station enters stage 2 *during* a transmission cycle (i.e., bc_1 is counted down to zero using the second method in Section V-A), it will set the initial value of bc_2 to 0.

On the other hand, if a station enters stage 2 at the *beginning* of a transmission cycle (i.e., bc_1 is less than or equal to zero after reducing by K, using the first method in Section V-A), it will generate an initial value for bc_2 uniformly distributed over the interval [0, CW2]. CW2 is contention window for the second stage. It has minimum value $CW2_{min}$ and maximum value $CW2_{max}$. The initial value of CW2 is $CW2_{min}$.

After the channel has been idle for DIFS duration, each station in stage 2 decreases bc_2 by 1 after each slot. When bc_2 reaches zero and the channel is idle, the station will begin its transmission. Before the bc_2 of a station reaches zero, if a frame sent by some other station is successfully received or overheard (e.g., RTS or CTS frames in the case of the RTS/CTS access method), the former station will return to stage 1 and double its CW1.

When a collision[3] happens, the colliding stations will double their CW2, and generate a new bc_2 value from the interval [0, CW2]. They will remain in the second stage and repeat the above channel contention procedure for stage 2 until someone wins the channel.

After a station finishes a successful transmission, its CW2 will be halved and it will return back to the *first* stage. The reason we choose to halve CW2 after each successful transmission is because the range of M (the number of stations in stage 2) is small compared to N (total number of active stations in the network).

Figure 4 illustrates an example with $CW1_{min} = 31$ and $K = 31$. At the beginning of a transmission cycle, at time t0, all stations reduce their bc_1 by K. Since stations 3 and 5 then have $bc_1 \leq 0$, they enter stage 2 and contend for channel access. Stations 1, 2 and 4 stay in stage 1 without participating in channel contention. However, after 4 *idle* slots, at time t1, bc_1 of station 2 is counted down to zero and it enters stage 2 also (interference happens here). From

[3] When an RTS is not followed by a CTS, or Data is not followed by an ACK, a collision is assumed to have occurred.

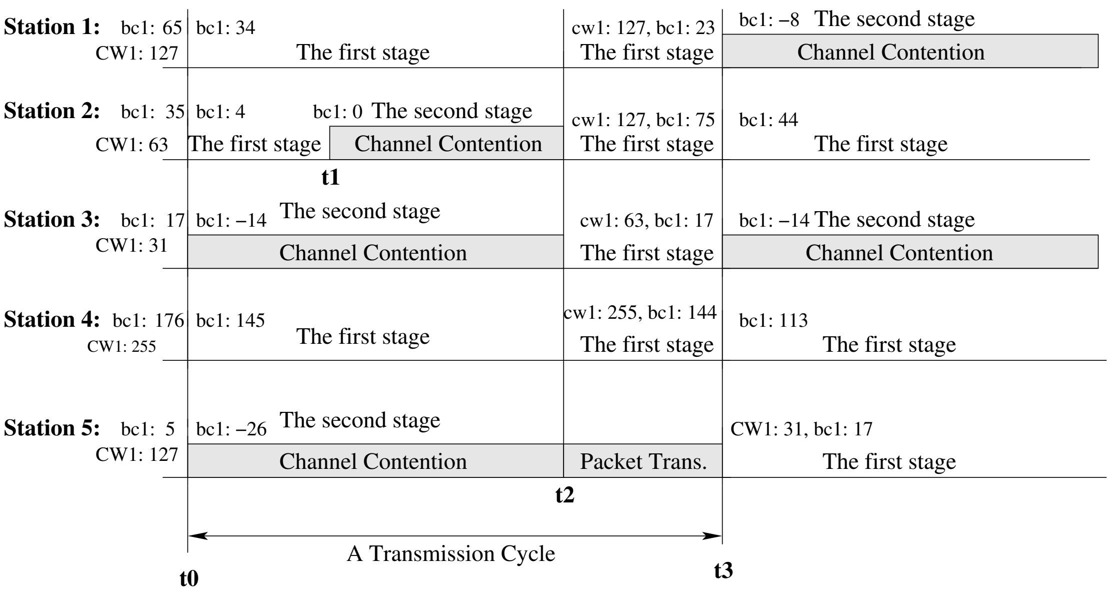


Fig. 4. The first stage of DSCR

time t1 to t2, stations 2, 3 and 5 continue to contend for the channel following the procedure for the second stage. At time t2, when station 5 wins the channel, stations 2 and 3 double their CW1 and return back to the *first* stage. They also generate a new bc_1 value which is uniformly distributed over the interval [0, CW1]. At time t3 when station 5 finishes its transmission, a new transmission cycle begins. Station 5 resets CW1 to $CW1_{min}(CW1_{min} = 31)$, returns back to stage 1 and generates a new value for bc_1 from the interval [0, CW1]. At this time, all stations except for station 5 reduce their bc_1 by K (since station 5 just finished transmitting a packet). Stations 1 and 3 now have $bc_1 \leq 0$, and thus would begin to contend for the channel in the new transmission cycle.

C. Special case with a single flow

When there is only a single flow in the network, DSCR will perform similar to IEEE 802.11 DCF. We now explain this using the example in Figure 5. At time t0 in Figure 5, at the start of a new transmission cycle, station 0 has the only flow in the network and its bc_1 is 45. Because its bc_1 is more than K (K = 31), after reducing bc_1 by K, station 0 will stay in stage 1. Since only station 0 has packets to transmit, there is no other station in stage 2. After a DIFS duration, station 0 begins to decrement its bc_1 by 1 after each idle slot. At time t1, bc_1 reaches zero and station 0 begins to transmit. At time t2, when station 0 finishes its transmission, a new transmission cycle begins. Station 0 resets CW1 to $CW1_{min}(CW1_{min} = 31)$ and generates a new value, say 21, for bc_1. Since it just finished transmitting a packet, station 0 will not reduce bc_1 by K at time t2. Then, after a DIFS duration and 21 idle slots, the bc_1 of station 0 will again be counted down to zero, and station 0 will transmit.

Clearly, if $CW1_{min}$ of DSCR has the same value as CW_{min} of 802.11, DSCR will perform exactly the same as IEEE 802.11 DCF when there is only one flow in the network.

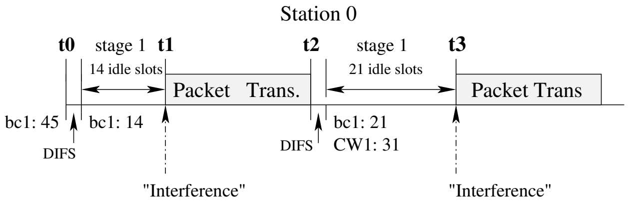


Fig. 5. Interference between the two stages

VI. Performance Evaluation

In this section, we present the simulation results for DSCR. All the simulation results are based on a modified version of ns-2 network simulator [14]. The channel bit rate is set to 11 Mbps. Physical layer preamble and header length is set to 192 μs according to IEEE 802.11 standard (with Direct Sequence Spread Spectrum) [2]. Unless mentioned otherwise, the packet payload size used is 512 bytes and the RTS/CTS access method is used. We use *Constant Bit Rate* traffic and traffic rate is aggressive enough to keep an active station backlogged.

For DSCR, we set K to 31, $CW1_{min}$ to 31, $CW1_{max}$ to 1023, $CW2_{min}$ to 3 and $CW2_{max}$ to 1023 unless mentioned otherwise. The impact of these parameters on

DSCR performance will be discussed in the Section VI-B below.

Initially, we perform simulations for scenarios where no hidden terminals exist. Later, in Section VI-D, we consider the case of hidden terminals.

A. DSCR, 802.11 and MACAW with RTS/CTS

A.1 Stable Traffic

We place all stations at the same location (hence no hidden terminals), and increase the number of active stations (denoted as N) from 1 to 256. The performance of DSCR, 802.11 and MACAW is compared in terms of aggregate throughput over all active stations. In each case, the number and traffic rates of source stations remain unchanged for the entire simulation. The simulation lasts for 30 seconds for each run and the results are averaged over 10 runs.

Taking into account the overhead introduced by data packet header, RTS/CTS frames, physical layer preamble and header, 28.86% of the total transmission time is used to transmit the payload (512 byte packets). Therefore, without the cost of channel contention resolution, ideally we can expect the maximum aggregate throughput of 3174.7 Kbps.

The simulation results of DSCR, IEEE 802.11 and MACAW are presented in Figure 6(a). The maximum aggregate throughput we can ideally obtain is also plotted in Figure 6(a). Generally, DSCR achieves better throughput than 802.11 and MACAW. The only exception occurs when there are only 1 or 2 flows in the network. In these two cases, MACAW performs better than DSCR because of the small $CW_{min} (CW_{min} = 3)$ for MACAW. In the case of a single flow, DSCR behaves exactly the same as 802.11 (as explained in Section V-C). The average number of idle slots between two consecutive transmissions is 16 for both 802.11 and DSCR. Consequently, they have lower throughput (2515 Kbps) than MACAW (3036 Kbps) for a single flow network.

With two or more flows in the network, the second stage of DSCR begins to take effect. As we show in Section VI-B below, when N is increased up to 256, there are less than 26 stations entering the second stage in each transmission cycle. Since the number of stations in stage 2 is small and has less variation compared to N, DSCR exhibits higher stability than 802.11. The performance gap between DSCR and 802.11 increases for larger values of N. Particularly, when N is 256, the aggregate throughput of DSCR is 56% more than 802.11.

DSCR achieves better stability than 802.11 by reducing the number of stations contending for the channel, while MACAW achieves stability by slowly decreasing the contention window. Therefore, DSCR does not suffer from the unnecessarily large CW value as MACAW does, and thus performs well in both small and large networks. In a moderate size network with N = 16, the aggregate throughput of DSCR is 46% more than MACAW.

We repeat the above simulations for various payload sizes (256 and 2048 bytes). Results are presented in Figure 6(b) and 6(c). Comparing the results in Figure 6, we observe DSCR, 802.11 and MACAW follow a common trend for various payload sizes. However, the performance gap between DSCR and 802.11 increases with a smaller packet size since the channel contention is associated with each packet's transmission. With a smaller packet size, the same amount of data will require more packets, implying an increase in the cost of contention. Particularly, with 256 active stations, DSCR achieves 64% more aggregate throughput over 802.11 for 256 byte packets and 33% more for 2048 byte packets.

A.2 On-off Traffic

Now we consider the case where the network load changes with time. In particular, the network load changes every 5 seconds. The number of active stations for each interval is shown in Figure 7(a). The queue size for each station is set to 2 so that the queue is drained soon after the traffic source is turned off. The throughput is measured over 1 second periods and the results from DSCR, 802.11 and MACAW are plotted in Figure 7(b).

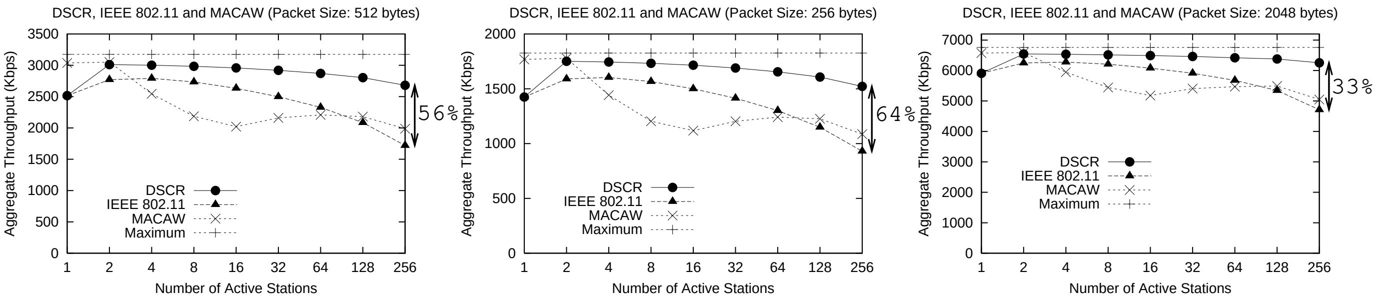
DSCR achieves higher throughput than both 802.11 and MACAW except for cases when there are only 1 or 2 flows in the network. This is consistent with the results in Section VI-A.1 with stable traffic. Moreover, the aggregate throughput of DSCR exhibits reasonably good stability throughout the entire simulation, even with changes in the network load.

MACAW also achieves better stability over 802.11 in the sense that its throughput is less sensitive to the network size. However, it sacrifices the throughput at moderate size networks. It is interesting to observe that MACAW has larger oscillations over short time intervals.

B. More about DSCR

In this subsection, we study the behavior of DSCR more carefully.

Recall that in DSCR, on losing the channel contention in stage 2, CW1 is doubled; and CW1 is reset to $CW1_{min}$ after each successful transmission. In Figure 8(a), we plot the CW1 distribution for a particular station in the stable traffic situation (as in Section VI-A.1) with N = 128. Horizontal axis of Figure 8(a) represents simulation time lasting from 0 to 30 seconds. Y-axis indicates the value



(a) Packet Size: 512 bytes (b) Packet Size: 256 bytes (c) Packet Size: 2048 bytes

Fig. 6. DSCR, IEEE 802.11 and MACAW with various packet sizes (horizontal axis is plotted in log-scale)

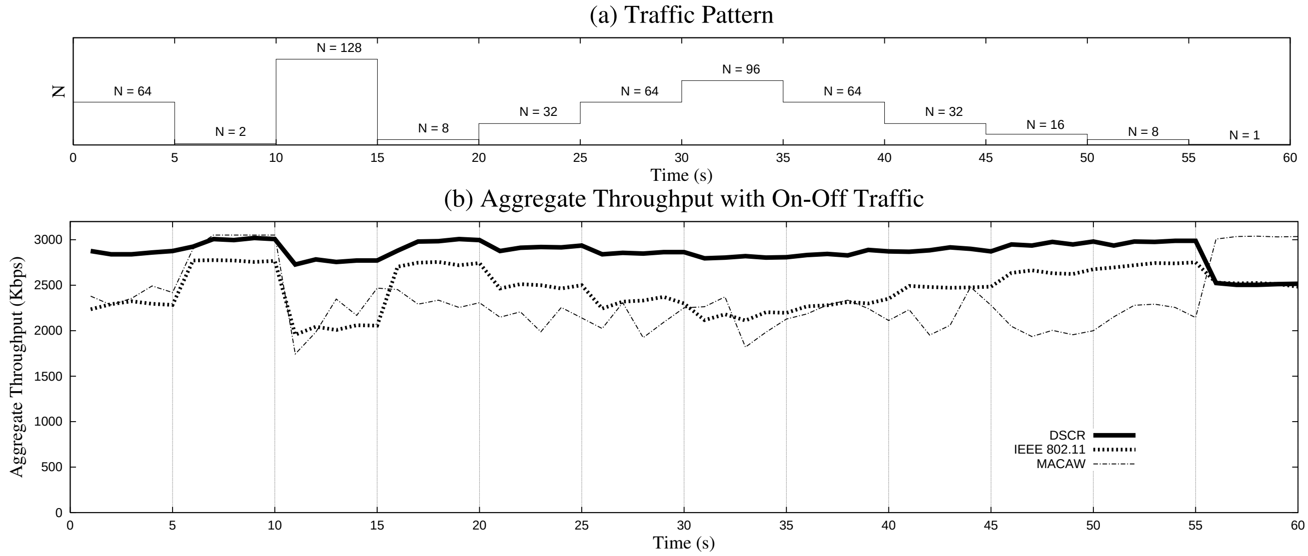


Fig. 7. DSCR, 802.11 and MACAW with On-Off traffic

of CW1. Whenever CW1 is changed, the corresponding point for CW1 is drawn. For example, each point where $CW1 = 31$ indicates when the station just finished a successful transmission and reset CW1 to 31. The distribution of CW1 affects the number of stations entering stage 2. Since bc_1 is uniformly distributed over the interval [0, CW1], and only stations with $bc_1 \leq K$ can enter stage 2 at the beginning of each transmission cycle, large values of CW1 help reduce the chance of entering stage 2. We can see from Figure 8(a) that CW1 spans over a wide range and has the value $CW1_{max} = 1023$ frequently. This implies that this station does not enter stage 2 very often in such a highly loaded network (N = 128). Consequently, out of all stations, the number of stations in stage 2 in each transmission cycle is relatively small. On the other hand, this station does get chances to transmit from time to time (each point with CW1 = 31 indicates a successful transmission).

The average number of active stations in stage 2 for each transmission cycle is plotted in Figure 8(b). When N is 128, there are less than 11 stations in stage 2 on average. Since only the stations in stage 2 may try to transmit a packet, the number of collisions in DSCR is lower than 802.11. Using the RTS/CTS access method with ideal channel condition, each collision leads to an RTS retransmission. We count the average number of RTS retransmissions per second for DSCR and 802.11 and plot the results in Figure 8(c). The difference is significant. With 128 active stations, there is an average of 1090 RTS retransmissions each second using 802.11 and only 310 for DSCR. The simulation results reveal that through two stage contention resolution, DSCR reduces channel contention significantly. Channel utilization has been improved and the MAC protocol exhibits higher stability in the sense that the throughput is less sensitive to the network size. In a small network, DSCR gains over 802.11 by using a smaller value of CW2 for the second stage ($CW2_{min} = 3$). Notice that if 802.11 uses the same small value for CW, its performance would degrade even faster with the increase of network size (as seen in Figure 3). Thus, except for the case of a single flow, in which DSCR has the same throughput as 802.11, DSCR achieves better and more stable channel utilization.

Next, we identify the impact of the parameter K on the performance of DSCR. We define "Interference Ratio" for each transmission cycle as the number of stations in stage 2 causing interference, divided by the total number of stations in stage 2. The average interference ratio with various K is presented in Figure 9(a). We can see that larger the K, lesser the interference between the two stages. When K is 31, the interference only happens to about 11% of the stations in stage 2 for large networks and DSCR achieves the best aggregate throughput. This is shown in Figure 9(b). For various values of K, the aggregate throughput gap is greater when the network is small. This is because, the number of stations entering stage 2 depends more on the choice of K when N is small. With a smaller value of K, the chance that no station enters stage 2 is larger. When such situations occur, the channel will be idle until interference happens, which leads to wasted channel time in the idle state.

Notice that when K is 0, the performance is the worst, but it is still better than 802.11 (compared to the curve of 802.11 in Figure 6(a)). When K is 0, the first stage of DSCR is functionally similar to the backoff procedure of 802.11. Better performance of DSCR with $K = 0$ is because that the second stage is used to resolve contention until some station wins.

Consider the other parameters of DSCR, namely, $CW1_{min}$, $CW1_{max}$, $CW2_{min}$ and $CW2_{max}$. A larger $CW1_{min}$ will help reduce the interference between the first and second stages. However, we do not want to degrade the single flow performance with respect to 802.11, so $CW1_{min}$ is chosen to be 31 (the same as the CW_{min} of 802.11). Since the number of stations in the second stage is usually small, $CW2_{min}$ should not be too large. Otherwise, the performance will suffer because of wasted channel time in the idle state. The simulation results show that 3 is an acceptable value for $CW2_{min}$. The choice of $CW1_{max}$ and $CW2_{max}$ have no major impact on the aggregate throughput of DSCR, provided they are large enough to accommodate the maximum network size. However, by setting $CW1_{max}$ to infinity, we do observe increased unfairness. With $CW1_{max} = \infty$, some unlucky stations might get a very large CW1 and it is hard for them to gain a chance to transmit again. In our simulations, we choose $CW1_{max}$ as 1023 and $CW2_{max}$ as 1023.

C. DSCR, IEEE 802.11 and MACAW with basic access method

We repeat the steady traffic simulation scenario in Section VI-A.1 using the basic access method (i.e., RTS/CTS handshake is not used). The simulation lasts for 30 seconds for each run and the results are averaged over 10 runs.

As shown in Figure 10, the performance gap between DSCR and 802.11 is now larger. This is primarily because of a larger collision cost. Since DSCR improves the performance through reduced collision probability, it will gain more with a larger collision cost. When N is 256, DSCR has 86% more aggregate throughput than 802.11 (This number is 56% when using the RTS/CTS access method).

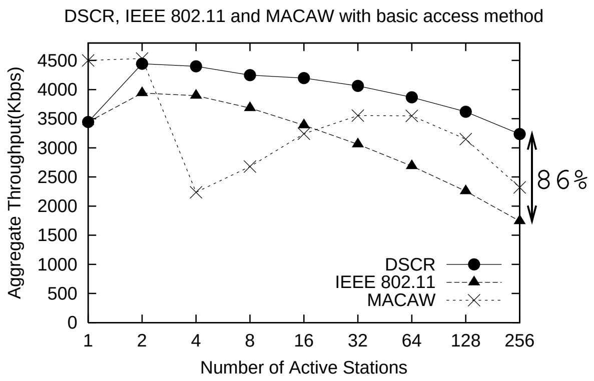


Fig. 10. The Basic Access Method (horizontal axis is plotted in log-scale)

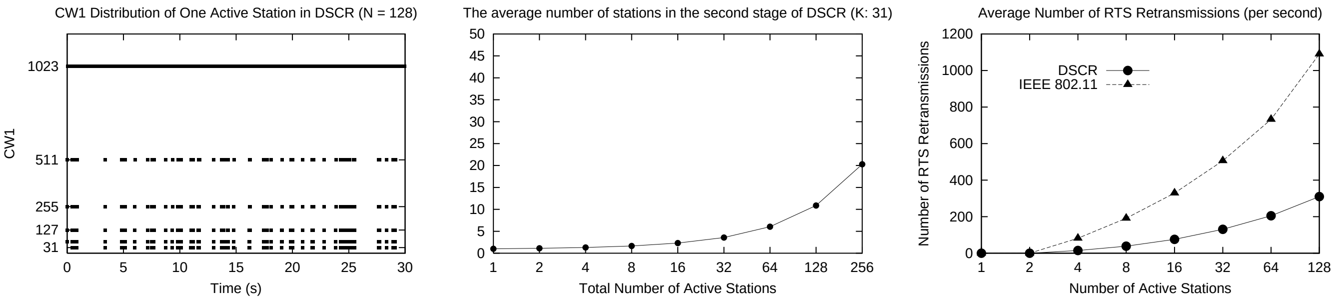
D. DSCR, IEEE 802.11 and MACAW with hidden terminals

This scenario is designed to simulate an infrastructure network with many hidden terminals. Specifically, the active stations are divided into two groups. We modified the simulator such that these two groups are hidden from each other. That is, stations in one group cannot sense any station in the other group, and vice versa. All active stations send packets (512 byte payload size) to the base station. Stations belonging to the same group are located at the same position. Each individual flow in this scenario has 50% of the total flows hidden from itself. The simulation lasts for 30 seconds for each run and the results are averaged over 10 runs.

Note that there are primarily two categories of collisions, which are caused by two different reasons:

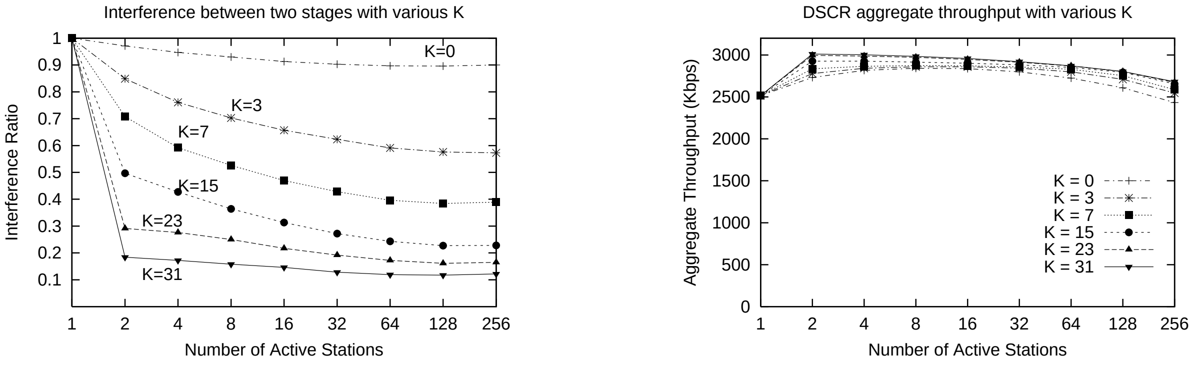
1. Type A: The contention resolution algorithm may schedule multiple stations to transmit at the same time (even under ideal channel condition).

2. Type B: Due to the limitation of the physical channel condition (e.g., hidden terminals), a station erroneously



(a) CW1 distribution with 128 active stations (b) Average number of stations in the second stage (c) Average number of RTS Retransmissions (per second)

Fig. 8. The Properties of DSCR (horizontal axis is plotted in log-scale)



(a) Interference Ratio vs. K (b) Aggregate Throughput vs. K

Fig. 9. The impact of K on DSCR performance (horizontal axis is plotted in log-scale)

senses a busy channel as being idle.

Type B collisions will increase significantly in the presence of hidden terminals. For example, stations 1 and 2 are hidden from each other and they both have packets backlogged for station 0. When station 1 is transmitting an RTS frame to station 0, station 2 will sense channel as being idle and begin its transmission, hence, causing the RTS to collide at station 0. Such a collision will not occur often without hidden terminals.

Without relying on black burst or other equivalent mechanisms, DSCR tries to reduce type A collisions. Without hidden terminals, previous simulation results have shown that DSCR performs better than 802.11 DCF and MACAW through improved contention resolution efficiency. The simulation results in Figure 11 show that DSCR preserves improved contention resolution efficiency even with hidden terminals.

Since type B collisions increase significantly with hidden terminals and they occur in all three MAC protocols, DSCR, 802.11 and MACAW all exhibit a decrease in aggregate throughput compared to Figure 6(a). Also, with the increase of type B collisions, the ratio of type A collisions as a fraction of the total collisions gets smaller. Even though DSCR still reduces type A collisions, the overall improvement is smaller. When N = 256, DSCR performs 19% better than 802.11 DCF.

With a moderate size network, MACAW suffers even more from the slow decrease of CW since type B collisions contribute greatly to the exponential increase of CW, which has nothing to do with channel congestion (e.g., the case of N = 2).

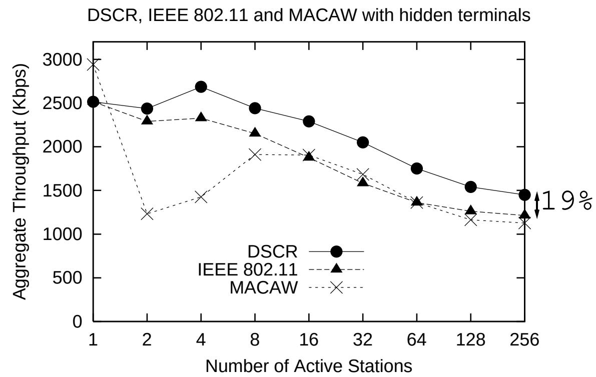


Fig. 11. DSCR, 802.11 and MACAW with hidden terminals (horizontal axis is plotted in log-scale)

VII. Analysis for The First Stage of DSCR

In analyzing the behavior of the first stage, we are interested in the average number of stations entering into the second stage for DSCR. To simplify the analysis, we make following assumptions:

- We approximate the first stage of DSCR by using p_2 to represent a station's probability of entering stage 2. This probability is directly related to the value of bc_1, which, in turn, is uniformly distributed over the interval [0, CW1]. Recall that in DSCR, by resetting CW1 to $CW1_{min}$, a station which just finished transmitting a packet will enter stage 2 within the next two transmission cycles if $CW1_{min} \leq K$. In the analysis, we simply assume p_2 is reset to 1 after a successful transmission, which will lead to slightly higher analysis results for the number of stations in stage 2 compared to the simulation results.
A station which fails to win channel access in stage 2 will double CW1. This is analogous to halving p_2.
- We assume that in steady state each active station behaves independently and has an identical distribution of p_2. Based on the stationary distribution of p_2, we obtain $E[p_2]$ and use this value as the average probability of entering stage 2 in steady state.
- We assume that no interference occurs between the two stages. Although not accurate, such an approximation can only introduce minor discrepancies since the interference ratio has already been shown to be small when K is 31 (refer to Section VI-B).

Assume that there are a total of N active stations in the network. For $i \in [1, N]$, let I_i be a random variable defined as follows:

$$I_i = \begin{cases} 1 & \text{if the } i^{th} \text{ station enters stage 2} \\ & \text{in a transmission cycle} \\ 0 & \text{otherwise} \end{cases}$$

and W_i be a random variable defined as:

$$W_i = \begin{cases} 1 & \text{if the } i^{th} \text{ station wins channel in stage 2} \\ 0 & \text{otherwise} \end{cases}$$

Let us consider the k^{th} station ($k \in [1, N]$). Note that the greater the number of stations entering the second stage, the smaller the probability of winning the channel access for a particular station in the second stage. Let M represent the number of stations in the second stage for a transmission cycle. Each of these M stations has an equal chance of winning channel access and one of them will eventually win (even though collisions and retransmissions may happen in the second stage). Conditioning on $I_k = 1$ and $M = j$, the probability that the k^{th} station wins channel access is: Prob$\{W_k = 1 \mid M = j, I_k = 1\} = \frac{1}{j}$.

With the k^{th} station in stage 2 (i.e., $I_k = 1$), the value of M can be from 1 to N with a total of N stations in the network. Since each station has the probability of $E[p_2]$ to enter stage 2, the value of M follows binomial distribution and can be represented as:

$$\begin{aligned} &\text{Prob}\{M = j \mid I_k = 1\} \\ &= \begin{pmatrix} N-1 \\ j-1 \end{pmatrix} \times E[p_2]^{j-1} \times (1 - E[p_2])^{N-j} \end{aligned}$$

where $j \in [1, N]$. Now, we have

$$\begin{aligned} &\text{Prob}\{W_k = 1 \mid I_k = 1\} \\ &= \sum_{j=1}^{N} [\text{Prob}\{W_k \mid M = j, I_k = 1\} \\ &\times \ \text{Prob}\{M = j \mid I_k = 1\}] \end{aligned}$$

Let this probability be denoted as p_w. Simplifying the above equation, we obtain

$$p_w = \text{Prob}\{\ W_k = 1 \mid I_k = 1\} = \frac{1 - (1 - E[p_2])^N}{N \times E[p_2]} \tag{1}$$

The next step is to obtain $E[p_2]$. Since p_2 has a maximum value of 1 after each successful transmission, and it will be halved after losing channel contention in stage 2, we can represent p_2 as:

$$p_2 = \frac{1}{2^c} \tag{2}$$

Here, $c \in [0, S]$ and $\frac{1}{2^S}$ represents the minimum value of p_2. Note that in our implementation of DSCR, with $CW1_{min}$ as 31 and $CW1_{max}$ as 1023, S = 5 (i.e., the maximum value of c is 5).

In this way, the first stage of DSCR can be modeled as a discrete-time Markov chain with $S+1$ states. The discrete state space now consists of the set of integers c=$\{$0, 1, 2, ... ,S$\}$. The state-transition matrix P is independent of time and is shown in Figure 12. In the state transition matrix, $1 - p_2$ accounts for the probability of not entering stage 2. In this case, c remains unchanged. $p_w p_2$ is the probability of entering stage 2 and winning the channel, where p_w is defined in Equation (1). With this probability, c will be reset to zero. $(1 - p_w)p_2$ accounts for the probability of entering stage 2 but failing to win the channel. With this probability, c will be increased by 1 until it reaches S.

The Markov chain is irreducible and has finite states, its stationary distribution μ exists and satisfies $\mu = \mu P$. We obtain μ as:

$$\mu_i = \begin{cases} p_w & \text{if } i = 0 \\ (1 - p_w)^i \times p_w & \text{if } 0 < i < S \\ (1 - p_w)^S & \text{if } i = S \end{cases}$$

State Transition Matrix P
Prob {c = j | c = i}

i\j	0	1	2	3		S
0	$(1-p_2) + p_w p_2$	$(1-p_w)p_2$	0	0	……	0
1	$p_w p_2$	$(1-p_2)$	$(1-p_w)p_2$	0	……	0
2	$p_w p_2$	0	$(1-p_2)$	$(1-p_w)p_2$	……	0
	……	……	……	……	……	……
S	$p_w p_2$	0	0	0	……	$(1-p_2) + (1-p_w)p_2$

Fig. 12. State Transition Matrix P for the first stage of DSCR

With the stationary distribution, $E[p_2]$ can be expressed as:

$$E[p_2] = \sum_{i=0}^{S} \frac{1}{2^i} \times \mu_i = 2p_w \frac{1 - (\frac{1-p_w}{2})^S}{1 + p_w} + (\frac{1 - p_w}{2})^S \tag{3}$$

Here $\frac{1}{2^i}$ represents the corresponding value of p_2 when $c = i$, as defined in Equation (2).

Equations (1) and (3) can be solved numerically to obtain $E[p_2]$. To see that there is a unique solution, notice that in Equation (1), when $N > 1$, $E[p_2] \in (0, 1)$ and $p_w \in (0, 1)$, p_w is a monotonically decreasing function of $E[p_2]$. When $E[p_2] = 1$, $p_w = \frac{1}{N}$. When $E[p_2]$ decreases to zero, p_w goes to 1. On the other hand, with p_w increasing from 0 to 1 in Equation (3), $E[p_2]$ will monotonically increase from $\frac{1}{2^S}$ to 1. Equations (1) and (3) have a unique intersection point which is the solution we desire. In the special case of N = 1, the solution is $p_w = 1$, $E[p_w] = 1$, which is also unique.

With $E[p_2]$ as the average probability of entering stage 2 and with the independence assumption among all stations, the number of active stations (M) in stage 2 for each transmission cycle follows binomial distribution. It has expected value as:

$$E[M] = N \times E[p_2] \tag{4}$$

Recall that S is the maximum value of c as defined in Equation (2). Varying S from 5 to 10, we calculated $E[p_2]$ from Equations (1) and (3). Consequently, the value of E[M] can be obtained from Equation (4). Figure 13 shows E[M] as a function of N. The E[M] measured by simulations for $N \leq 256$ and $CW1_{max} = 1023$ (i.e., $S = 5$) is also presented in Figure 13. While analysis results have a slightly higher E[M] than our simulation results, they agree with the general trend for E[M].

From the analysis results, we can see that when N is less than 256, the value of S does not affect E[M] much and E[M] is less than 26. When the network size increases further, a larger value of S (i.e., larger value of $CW1_{max}$) would be necessary. The analysis results show that when N is 512, E[M] is less than 35 with S being 6. When S is 10, E[M] can stay below 65 for N = 2048.

In summary, both analysis and simulation results show that DSCR effectively controls the number of stations entering stage 2 in each transmission cycle.

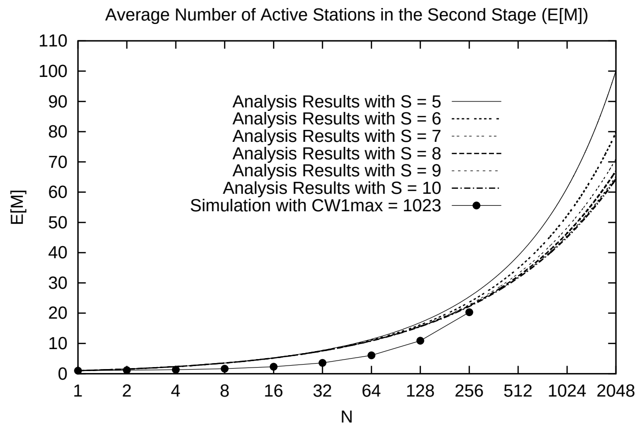


Fig. 13. E[M] vs. N (horizontal axis is plotted in log-scale)

VIII. Improving DSCR fairness

We have shown in Section VI that DSCR achieves significant aggregate throughput improvement over IEEE 802.11 DCF, especially in large networks. However, a close look at the throughput from each individual flow indicates that the fairness of DSCR is worse than IEEE 802.11. The degraded fairness of DSCR is primarily due to following two reasons:

- A station exponentially increases its CW1 whenever losing the second stage channel contention, but it linearly decreases bc_1 ($bc_1 \in [0, CW1]$) by K after each transmission cycle. In this way, an unlucky station with large bc_1 will not be able to enter the second stage until multiple transmission cycles later. For example, an unlucky station increases CW1 to 1023. With $bc_1 = 775$ and $K = 31$, it could take 25 transmission cycles for this station to enter the second stage. Consequently, the winning station tends to win multiple transmission cycles in a row.

- In DSCR, a small value of $CW2_{min}$ ($CW2_{min} = 3$) is used to help improve the throughput of small networks. However, since a winning station tends to win multiple transmission cycles in a row (as explained above), it is likely for this station to have a very small value of CW2 (Recall that CW2 is divided by 2 after each successful transmission). As a result, even the unlucky station with large bc_1 enters the second stage eventually, the winning station still has greater chance to win again.

In view of these two reasons, we modify DSCR to improve the fairness. For the rest of the paper, the modified version of DSCR is referred to as DSCR-F. The following modifications are made:

1. Instead of the linear decrement of bc_1 in DSCR, DSCR-F adopts non-linear decrement for bc_1. Specifically, let tc represent the number of transmission cycles since a station begins to stay in the first stage. At the end of each transmission cycle, the station deduct its bc_1 by $F(tc)$. While there are various choices possible for $F(tc)$, we evaluate one definition of $F(tc)$:

$$F(tc) = 2^{tc-1}K$$

The longer a station stays in the first stage, more aggressively it will reduce its bc_1. This will give the stations in the 1st stage more chances to enter the 2nd stage. On the other hand, since more stations enter the 2nd stage, the channel contention in the 2nd stage could be higher.
A different approach for non-linear decrement of backoff counter is also used by [18] in their fast collision resolution algorithm.

2. Intead of $CW2_{min} = 3$ in DSCR, DSCR-F set $CW2_{min}$ to 31, and the winning station will reset CW2 to $CW2_{min}$ after each successful transmission. In this way, we can see that the second stage of DSCR-F is quite similar to IEEE 802.11.

3. As we mentioned in Section VI-B, large value of $CW2_{min}$ has a negative impact on the performance of small networks. For example, for a network with two flows, most of the time there is only one flow entering the second stage. With $CW2_{min}$ = 31 and only one flow in 2nd stage, there is 16 idle slots between two consecutive transmissions on average. On the other hand, two flows will share the 16 idle slots in IEEE 802.11, resulting better throughput of 802.11. In DSCR-F, we make use of *interference* to improve performance of small networks by setting $CW1_{min}$ to 15 (Recall that DSCR sets $CW1_{min}$ to 31). K is also set to 15 accordingly. For the above example of two flows, there will be approximately 8 idle slots between two consecutive transmissions on average due to the interference.

We perform the same simulation as in Section VI-A.1 with payload packet size of 512 bytes. The fairness index, Max/Min throughput ratio and aggregate throughput of DSCR, DSCR-F, IEEE 802.11 and MACAW are reported in Figure 14.

Fairness index is defined as follows[19]:

$$\text{Fairness index} = \frac{(\sum_f Thr_f)^2}{N \times \sum_f Thr_f^2}$$

where Thr_f represents each individual flow's throughput. N represents the total number of flows.

Among all individual flows, Max/Min throughput ratio is the ratio of the maximum throughput over the minimum throughput.

From Figure 14, we can see that DSCR-F has achieved comparable fairness as 802.11, which is a significant improvement over DSCR. Even through DSCR-F has slightly degraded throughput in small networks compared to DSCR, DSCR-F preserves the stability of DSCR and the aggregate throughput of DSCR-F is much higher than 802.11 for networks with more than 16 flows. Notice the single flow throughput of DSCR-F is even higher than DSCR since the $CW1_{min}$ is smaller in DSCR-F. MACAW has better fairness than 802.11 and DSCR-F due to the help of *window copying mechanism*.

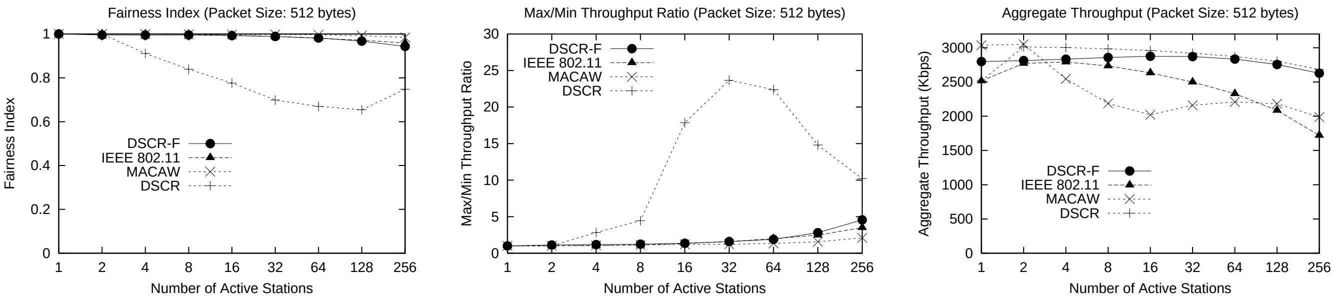
IX. Performance Comparison of DSCR-F and IEEE 802.11 in Random Topologies

To verify the improvement of DSCR-F over IEEE 802.11 DCF in real networks, we generated 30 different topologies for 80 stations in a 1000m $\times$ 1000m area. Each station picks one of its one hop neighbor (if there is any) to send packets to. The total number of flows varies from 70 to 75 depending on the chosen topology. All flows are always backlogged and the payload packet size is 512 bytes. The aggregate throughput of DSCR-F and IEEE 802.11 is reported in Figure 15(a). The same results are presented in Figure 15(b) in the form of throughput ratio (DSCR-F over 802.11). We can see that DSCR-F achieves 10% to 47% more throughput compared to 802.11 in these simulated random topologies.

DSCR-F gains more throughput by reducing the collisions among stations, which can be seen clearly from Figure 16. The average number of RTS retransmissions (per second) for DSCR-F and 802.11 are reported in Figure 16.

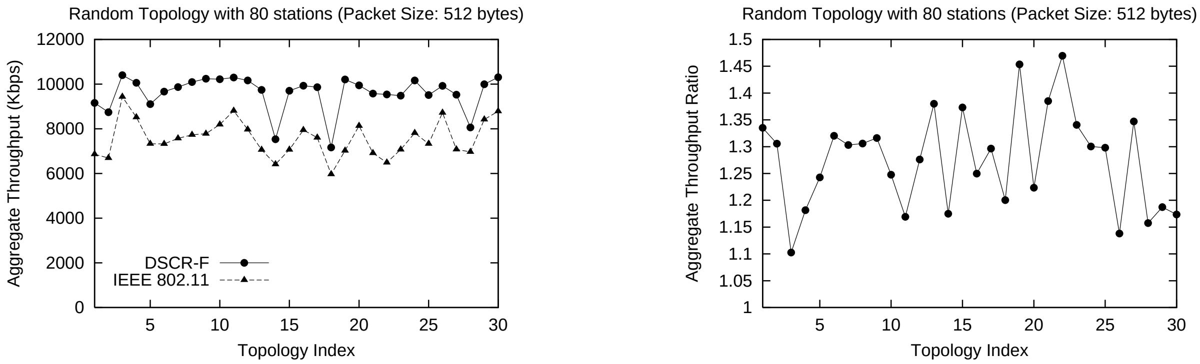
X. Conclusion

In this paper, we present a dual stage contention resolution MAC protocol (DSCR). DSCR is compatible with 802.11 and differs from 802.11 only in the backoff mechanism used for contention resolution. With the help of two "virtual" contention resolution stages, DSCR significantly reduces channel contention among active stations in highly loaded networks. Both simulation and analysis have shown that with a total 256 active stations, the average number of stations actually contending for the channel



(a) Fairness Index (b) Max/Min Throughput Ratio (c) Aggregate Throughput

Fig. 14. Fairness of DSCR-F, DSCR, IEEE 802.11 and MACAW



(a) Aggregate Throughput (b) Aggregate Throughput Ratio of DSCR-F over 802.11

Fig. 15. DSCR-F and IEEE 802.11 DCF in Random Topologies

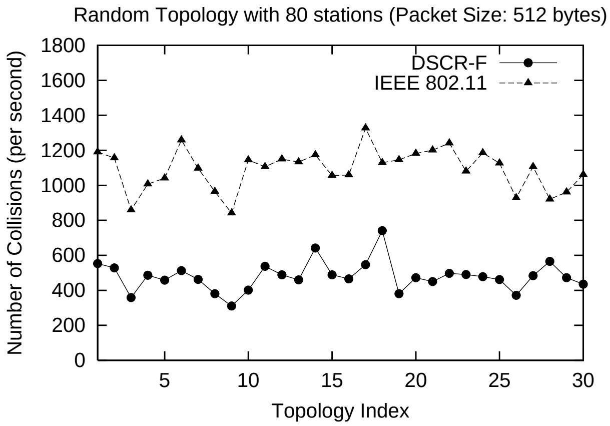


Fig. 16. The average number of RTS retransmissions of DSCR-F and 802.11

in each transmission cycle is less than 26. While maintaining relative stability with an increased network size, DSCR maintains better performance in small networks as well. Overall, without relying on blackburst or other equivalent mechanisms (e.g., as in HIPERLAN/I), DSCR achieves better channel scheduling efficiency and improved stability which are desired features for a MAC protocol. The improved contention resolution efficiency of DSCR is still preserved even in the presence of hidden terminals.

As a variation of DSCR, DSCR-F achieves fairness comparable to IEEE 802.11 while preserving much of the throughput gain over 802.11. Simulation results show that DSCR-F achieves more aggregate throughput than 802.11 in random topologies.

REFERENCES

[1] Leonard Kleinrock and Fouad A. Tobagi, "Packet Switching in Radio Channels: Part I- Carrier Sense Multiple-Access Modes and Their Throughput-Delay Charateristics," *IEEE Transactions on Communications*, vol. COM-23, no. 12, December 1975.

[2] IEEE, "Wireless LAN Medium Access Control(MAC) and Physical Layer(PHY) Specifications," *IEEE Standard 802.11*, June 1999.

[3] H. S. Chhaya and S. Gupta, "Performance modeling of asynchronous data transfer methods of IEEE 802.11 MAC protocol," *Wireless Networks*, pp. 217–234, 1997.

[4] B. P. Crow, I. Widjaja, J. G. Kim, and P. T. Sakai, "IEEE 802.11 wireless local area networks," *IEEE Communication Magazine*, pp. 116–126, Sep 1997.

[5] Y. C. Tay and K. C. Chuan, "A Capacity Analysis for the IEEE 802.11 MAC Protocol," *ACM/Baltzer Wireless Networks*, , no. 2, pp. 159–171, March 2001.

[6] J. Weinmiller, H. Woesner, J. P. Ebert, and A. Wolisz, "Analyzing and tuning the distributed coordination function in the IEEE 802.11 DFWMAC draft standard," *Proc. MASCOT, San Jose, CA*, Feb 1996.

[7] J. Weinmiller, M. Schlager, A. Festag, and A. Wolisz, "Performance study of access control in wireless LANs IEEE 802.11 DFWMAC and ETSI RES 10 HIPERLAN," *Mobile Networks and Applications*, pp. 55–67, 1997.

[8] G. Bianchi, "Performance Analysis of the IEEE 802.11 Distributed Coordinated Function," *IEEE Journal on Selected Areas in Communications*, vol. 18, no. 3, pp. 535–547, March 2000.

[9] G. Bianchi, L. Fratta, and M. Oliveri, "Performance Evaluation and Enhancement of the CSMA/CA MAC Protocol for 802.11 Wireless LANs," *Proceedings of PIMRC, Taipei, Taiwan*, pp. 392–396, Oct 1996.

[10] F. Cali, M. Conti, and E. Gregori, "IEEE 802.11 Protocol: Design and Performance Evaluation of an Adaptive Backoff Mechanism," *IEEE Journal on Selected Areas in Communications*, vol. 18, no. 9, pp. 1774–1786, September 2000.

[11] F. Cali, M. Conti, and E. Gregori, "Dynamic tunning of the IEEE 802.11 protocol to achieve a theoretical throughput limit," *IEEE/ACM Trans. on Networking*, vol. 8, no. 6, pp. 785–799, December 2000.

[12] ETSI TC-RES, "Radio Equipment and Systems(RES); HIgh PErformance Radio Local Area Network(HIPERLAN) Type 1; Functional specification," *European Telecommunication Standard ETS 300 652*, Oct 1996.

[13] Vaduvur Bharghavan, Alan Demers, Scott Shenker, and Lixia Zhang, "MACAW: A Media Access Protocol for Wireless LANs," *IEEE SIGCOMM*, 1994.

[14] VINT Group, "UCB/LBNL/VINT network simulator ns (version 2)," *http://www.isi.edu/nsnam/ns*.

[15] G. Anastasi, L. Lenzini, and E. Mingozzi, "Stability and Performance Analysis of HIPERLAN," *Proc. IEEE INFOCOM*, pp. 134–141, 1998.

[16] S. Chevrel, A. H. Aghvami, H. Y. Lach, and L. Taylor, "Analysis and optimisation of the HIPERLAN Channel Access Contention Scheme," *Wireless Personal Communications*, pp. 27–39, 1996.

[17] K. Fu, Y. J. Guo, and S. K. Barton, "Performance of the EY-NPMA protocol," *Wireless Personal Communications*, pp. 41–50, 1996.

[18] Y. Kwon, Y. Fang, and H. Latchman, "Design of MAC Protocols with Fast Collision Resolution for Wireless Local Area Networks," *submitted for publication*, 2002.

[19] R. Jain, G. Babic, B. Nagendra, and C. Lan, "Fairness, call establishment latency and other performance metrics," *Technical Report. ATM Forum/96-1173, ATM Forum Document*, August 1996.