Formulation of Distribution Coordination Function of IEEE 802.11 for Asynchronous Networks: Mixed Data Rate and Packet Size

Mustafa Ergen, Pravin Varaiya

{ergen,varaiya}@eecs.berkeley.edu

Department of Electrical Engineering and Computer Sciences

University of California Berkeley

Abstract

In CSMA/CA networks in which stations have different data rates, some stations are penalized because slow stations receive more time to transmit. Thus, a single low data rate station unfairly brings down the throughput of the high data rate stations. We introduce a simple and standard-compliant algorithm to utilize the channel fairly.

We first provide a formulation for the throughput with mixed data rate connections. To alleviate the low performance of high data rate stations, we introduce a mechanism that implements an adaptive scheme to adjust the packet size according to the data rate. With this scheme, stations occupy the channel for equal amounts of time. We then extend the scheme to a frame aggregation scheme to show how different packet sizes affect performance.

Index Terms

IEEE 802.11, Distributed Coordination Function, Fairness, QoS, Frame Aggregation, Wireless VoIP, Markov Model.

This research was supported by ARO-MURI UCSC-WN11NF-05-1-0246-VA-09/05 and National Semiconductor.

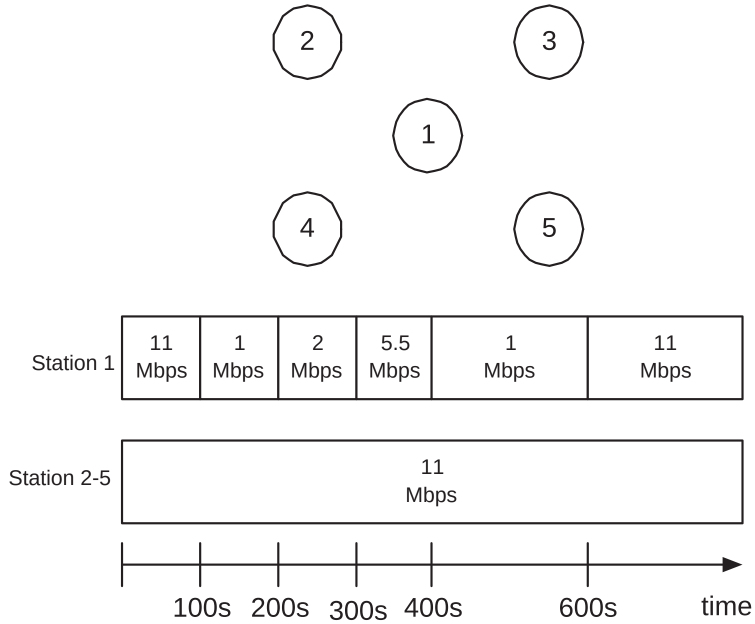

Fig. 1. Station's data rates throughout the simulation

I. Introduction

Wireless communications is evolving to a stage when each device will be a node in a mobile network with multiple interfaces. This also brings the growth of multimedia applications that impose requirements on communication parameters [1]. As a result, wireless networking is moving towards asynchronous connectivity.

We consider a scenario with several IEEE 802.11 [2]-[5] compliant nodes located near each other, which can transmit at different data rates and packet sizes. Each node may initiate packet transmission with variable physical transmission rates depending on its connection quality. Sometimes different IEEE standards may share the same spectrum with the same access mechanism as we see with IEEE 802.11b [3] and 802.11g [5] networks. Similarly, a station's packet size can also change during the connection depending on the type of flow. We are interested in obtaining analytical formulas for throughput in such a scenario. We suppose that every flow of a station is saturated, i.e., there is always a packet to transmit when a station has the right to transmit [6], [7].

Our approach begins by introducing the Markov chain model from [8], which in turn is an enhancement of Bianchi's model [6]. The performance differences between these two models are discussed in the next section; however, our analytical formulation can be applied to Bianchi's model as well.

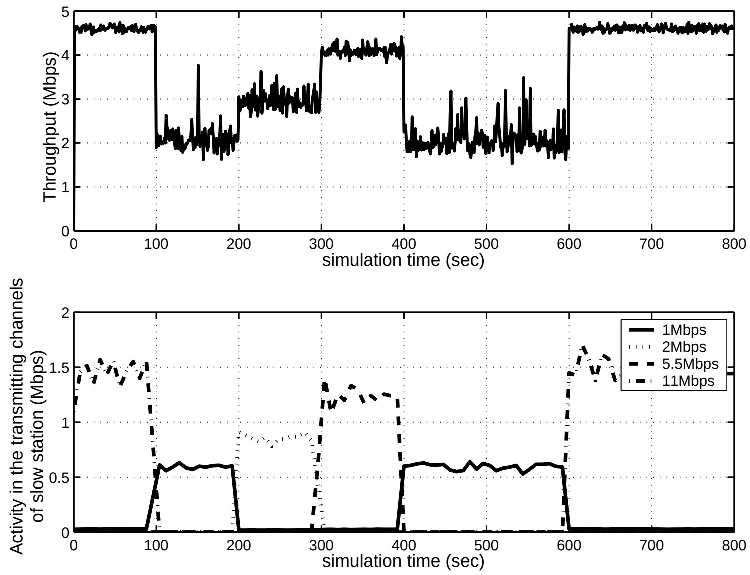

Fig. 2. Throughput in a mixed data rate environment

Case Study:

We simulate a scenario in *OPNET* in order to revisit the anomaly presented in [9]. The network has five stations, all with 11 Mbps data rate except for *station 1*, which changes its rate over time as in Figure 1. The upper plot of Figure 2 shows the total throughput of the network; the lower plot shows the activity in *station 1*'s channels. As seen, there are four channels each corresponding to one rate in IEEE 802.11b [3].

Station 1 severely impacts the network as seen in the upper plot of Figure 2. For example, between 100s and 200s (and again between 400s and 600s) *station 1* with *1 Mbps* data rate decreases the total throughput by more than half. Figure 3 shows plots of two stations, from which one may infer that the total throughput is divided equally, and stations with higher data rate experience the same throughput as the slow station. Recent papers [9], [10], [11], [12] have reported this behavior in IEEE 802.11 CSMA/CA networks.

The cause of this anomaly is buried in the basic *CSMA/CA* channel access method: Once a station acquires the transmission opportunity, it uses it as long as necessary to transmit a packet. As a result, if a station operates with a lower data rate it takes longer to transmit the same payload, leading to channel under-utilization.

In general, lower data rates are selected to increase robustness against interference; each transmitter selects the transmission rate appropriate to the wireless channel it faces. The selection mechanism is called rate adaptation;

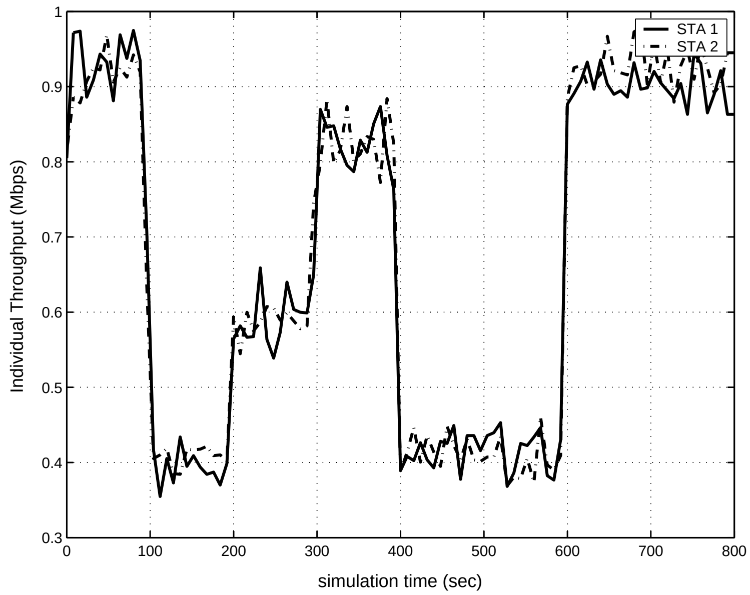

Fig. 3. Individual throughput in a mixed data rate environment

the mechanism is often proprietary. Several rate adaptation mechanisms are discussed [13], [14], [15].

We keep the rate adaption mechanism outside our focus, and assume that stations change their rate according to some mechanism. As we saw, stations with the lowest quality channel will consume more channel resources than those with better channel conditions. But in multi-rate networks, it is more efficient to equally share the channel *time* rather than the channel bandwidth [11]. In [11] the nodes' channel access is prioritized using different contention windows in order to guarantee higher throughput for higher rate nodes. One difficulty with this approach is that if the transmission probabilities are unequal, it is very hard to solve the Markov model since the probability of finding the channel busy p is not the same anymore [8], [16]. On the other hand a similar paper [17], prior to [11], guarantees equal temporal shares for all stations by allowing high rate stations to send bursts of frames in order to extend the time they occupy the channel.

Fairness is addressed in [18]-[26]. Fairness of DCF is investigated in [21], [22], with models that tune the network parameters with respect to channel and load conditions. The model in [20] introduces an adaptive fair protocol in the context of IEEE 802.11e.

In this paper, in contrast with [17], we couple data rate with packet size by a packet size adjustment scheme. We

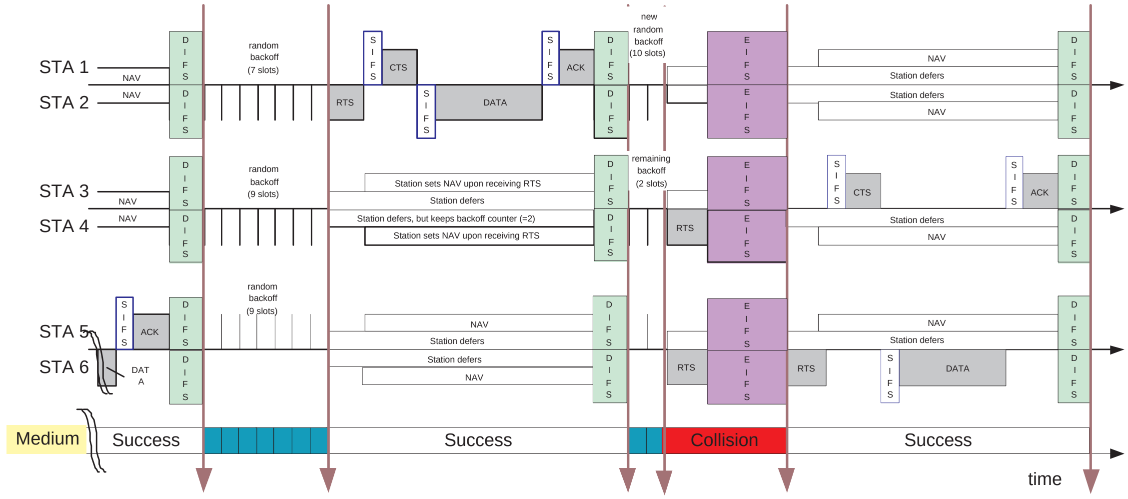

Fig. 4. An illustration of DCF Mechanism: STA 6 selects a backoff counter 0 and transmits immediately.

also investigate a packet aggregation scheme at the end of the paper. Our simple modification is effective and does not change the IEEE 802.11 standard. The modifications of [17] and [11] appear not to comply with the standard. This paper extends our previous work [12]: We contribute detailed simulation results and obtain the throughput formulation for different packet sizes and data rates, and analyze a frame aggregation scheme.

The paper is organized as follows. Section II describes the formulation for individual throughput and provides the basic analysis. The formulation for mixed data rate and mixed packet sizes occupies Section III. We discuss optimization schemes in Section IV and conclude the paper with Section V.

II. Throughput Formulation for Individual Stations

A typical medium utilization trace for a fully connected network in saturated traffic load is shown in Figure 4. We can say that a station is either in *transmission* or *reception* or *idle* state. In order to construct a Markov model for the behavior of a station we need to identify the events that trigger the state changes. Different models identify different events. The Bianchi model, which we call (802.11^b), defines an event as a *virtual slot*, which could be an *empty-slot* or a *transmission-plus-an-empty-slot*, which implies that after each event the backoff counter is decremented. Of course this 'clustering' assumes that there are no consecutive transmissions and each transmission, whether or not it is successful, is followed by at least one *empty-slot*.

There will be two consecutive transmissions if the zero backoff (i.e., *null* backoff) counter is selected immediately

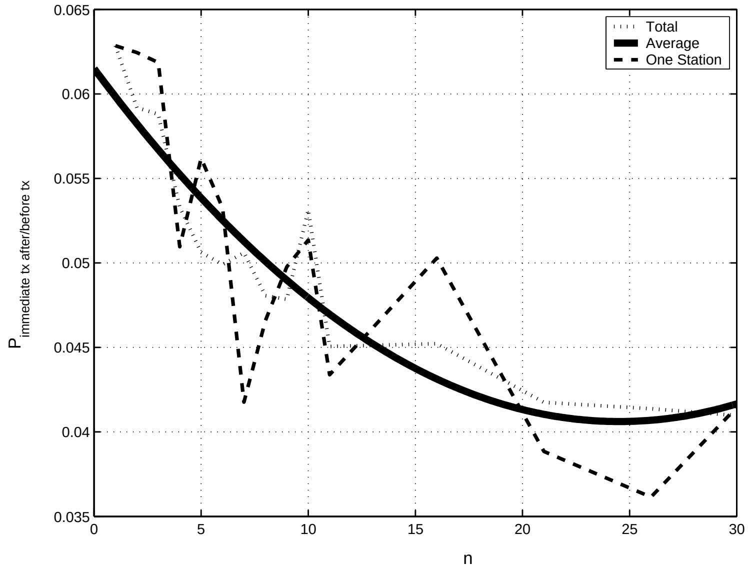

Fig. 5. Zero backoff probability

after a transmission. Figure 5 shows the probability of two consecutive transmissions, obtained with the *OPNET* simulation tool. As is evident from the figure, the probability decays as the number of stations increases; the probability is significant if the number of stations is small.

The model in [8], which we call 802.11^{+}, adopts a slightly different clustering in which a *virtual slot* is either an *empty-slot* or a *transmission*. This model permits consecutive transmissions. Figure 6 compares the virtual slots in the two models.

The wireless medium is modeled as a two-state process with probability p of being busy and $1-p$ of being idle. The states in the Markov model of each station are the states of its multi-level backoff counter, indexed by (i, j) for $0 \leq i < m$ and $0 \leq j < W_i$. (Here m is the number of backoff levels and W_i is the backoff window at level i, see Figure 7.) Denote the stationary distribution of these states by $b_{i,j}$. Then

$$1 = \sum_{i=0}^{m} \sum_{j=0}^{W_i - 1} b_{i,j}. \tag{1}$$

Since a station transmits in states of the form $(i, 0)$, the probability of transmission τ can be obtained after solving

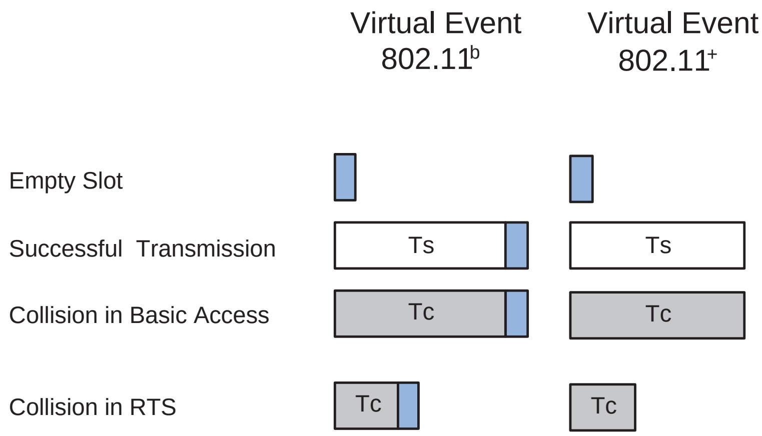

Fig. 6. Virtual events

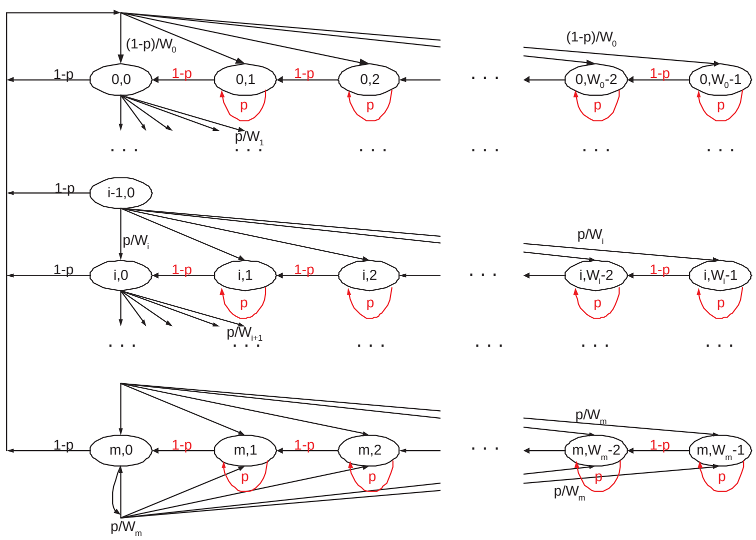

Fig. 7. Independent Markov model of 802.11$^+$ for multi-level backoff

the balance equations,

$$\tau = \sum_i b_{i,0} = \frac{1}{\frac{(1-2p)(W+1)+pW(1-(2p)^m)}{2(1-2p)(1-p)}}. \tag{2}$$

We assume that the stations are statistically independent [8], [6]. We can then find τ and p from (2) and $p = 1-(1-\tau)^{n-1}$, where n is the number of stations in the network.

Following [27], define P_{tr} as the probability that there is at least one transmission in the considered slot time

$$
\begin{aligned}
& \qquad BasicAccessMechanism \\
T_s(R^i,P) &= T_{DATA}(R^i,P) + SIFS + \delta + T_{ACK}(R^i) + \delta + DIFS \\
T_c(R^i,P) &= T^*_{DATA}(R^i,P) + \delta + EIFS \\
& \qquad RTS/CTSAccessMechanism \\
T_s(R^i,P) &= T_{RTS}(R^i) + SIFS + \delta + T_{CTS}(R^i) + SIFS + \delta \\
&\quad + T_{DATA}(R^i,P) + SIFS + \delta + T_{ACK}(R^i) + \delta + DIFS \\
T_c(R^i,P) &= T_{RTS}(R^i) + \delta + EIFS.
\end{aligned}
\tag{5}
$$

---

and P_s as the probability that a transmission is successful:

$$
\begin{aligned}
P_{tr} &= 1 - (1-\tau)^n, \\
P_s &= n\tau(1-\tau)^{n-1}.
\end{aligned}
\tag{3}
$$

Expressing the total throughput S as the ratio of successfully transmitted bits to the average length of a *virtual-slot* gives

$$
S = \frac{P_s E[P]}{(1-P_{tr})\sigma + P_s T_s(R^i) + (P_{tr} - P_s) T_c(R^i)}. \tag{4}
$$

Above, $E[P]$ is the average packet payload size; an *empty-slot* has duration σ and occurs with probability $(1-P_{tr})$; and there is a collision with probability $(P_{tr} - P_s)$. Also in equation (4), if the station has data rate R^i, $T_s(R^i)$ is the average time the channel is sensed busy due to a successful transmission, and $T_c(R^i)$ is the average time the channel is sensed busy by each station during a collision.

In (5), $T_{DATA}(R^i, P)$ is the time taken to send a packet of size P; $T_{RTS}(R^i)$, $T_{CTS}(R^i)$, $T_{ACK}(R^i)$ are the times taken to send the corresponding frames; $T^*_{DATA}(R^i, E[P^*])$ is the average time taken to send $E[P^*]$, which is the average length of the longest packet payload involved in a collision. (If all packets have the same size, $E[P] = E[P^*] = P$ [6].) The propagation delay is given by δ.

From an *OPNET* simulation, the parameters (P_{tr}, P_s, S) are estimated as follows:

$$
\begin{aligned}
P_{tr} &= \tfrac{\#Total\ ACK\ received + \#Collision}{\#Total\ ACK + \#Collision + \#Back-off\ slot}, \\
P_s &= \tfrac{\#Total\ ACK\ received}{\#Total\ ACK + \#Collision + \#Back-off\ slot},
\end{aligned}
\tag{6}
$$

while the throughput S is estimated as the successfully received packets divided by the simulation time. Figure 8

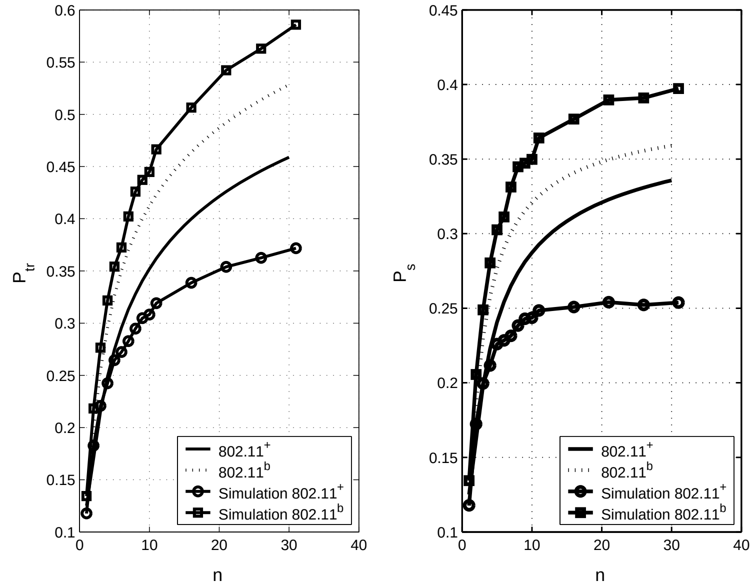

Fig. 8. P_{tr} and P_s for multi-level backoff

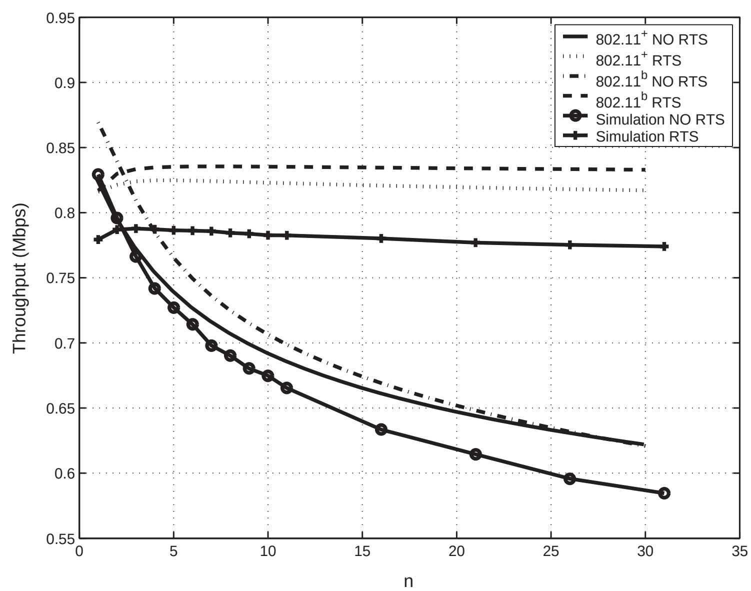

Fig. 9. Throughput for multi-level backoff, Data Rate= 1 Mbps

compares the analytical calculations for the 802.11^+ model with the simulation results.

From equation (4) we obtain the duration values using S, P_{tr}, and P_s from the simulation:

	$Basic$	RTS/CTS
T_s	$0.0088sec$	$0.0090sec$
T_c	$0.0088sec$	$0.0007sec$

To estimate P_{tr} and P_s for the 802.11^b model by simulation, we note that because of the large number of stations, consecutive transmission probability is low. As a result, equation (6) is modified for 802.11^b as follows:

$$\begin{aligned} P_{tr}^b &= \tfrac{\#Total\ ACK\ received + \#Collision}{\#Back-off\ slot} \\ P_s^b &= \tfrac{\#Total\ ACK\ received}{\#Back-off\ slot} \end{aligned} \tag{7}$$

Since each *back-off slot* for 802.11^b is either a *transmission-plus-empty-slot* or an *empty-slot*, it is enough to consider only the number of back-off slots for the total number of events. Figure 8 shows that 802.11^b underestimates the probabilities; on the other hand, 802.11^+ overestimates the probabilities.

Figure 9[1] shows the throughput performance. As one can see, the 802.11^+ model better matches the simulation, especially for small numbers of nodes. Note that the plots for 802.11^+ and 802.11^b in Figure 8 are not directly comparable, because the definitions of P_{tr} and P_s are different. The plots of the two models in Figure 9 are comparable as they refer to the same quantity, throughput.

We can find the throughput in a different way. Suppose the stations operate at different data rates. Consider the first K events for a network with n nodes which take time T. If L is number of empty slots, M is the number of successful transmissions and Y is the time wasted in collisions,

$$T = L\sigma + \sum_{i=1}^{n} X(i) T_s(R(i)) + Y, \tag{8}$$

where $X(i)$ as the number of successful transmission and of *station* i whose data rate is $R(i)$, and $\sum_{i=1}^{n} X(i) = M$.

[1]For this figure, we modified the Bianchi model [6] to obtain the plots for 802.11^b since the formulas for duration values (T_s and T_c) in [6] lack the *empty-slot* time after a transmission and T_c lacks the EIFS period after a collusion. Without these necessary modifications, the 802.11^b model would give higher throughput.

S_1	S_2	S_3	S_4	S_5	S_6	S_7	S_8
R^1	R^2	R^1	R^3	R^4	R^1	R^4	R^1
$T_s(R^1)$	$T_s(R^2)$	$T_s(R^1)$	$T_s(R^3)$	$T_s(R^4)$	$T_s(R^1)$	$T_s(R^4)$	$T_s(R^1)$
$T_c(R^1)$	$T_c(R^2)$	$T_c(R^1)$	$T_c(R^3)$	$T_c(R^4)$	$T_c(R^1)$	$T_c(R^4)$	$T_c(R^1)$

Fig. 10. Duration time and rate relation for $N = 8$ and $D = 4$

If $P(i)$ is the packet size, the throughput S_i of station i is

$$S_i = \frac{X(i)P(i)}{T}. \tag{9}$$

To calculate S_i we need to find $\frac{X(i)}{M}$ since we already know that the total throughput S is

$$S = \frac{ME[P]}{T} \tag{10}$$

If there are M successful transmissions and stations have equal chance of transmission, $X(i) = \frac{M}{n}$. This also follows from the assumption that the *virtual-slot*s are independent. In each *virtual-slot*, the probability of it being empty is $(1-\tau)^n$. Thus, $L = K(1-\tau)^n$. Similarly, the probability of finding a slot occupied by a successful transmission of *station* i is $\tau(1-\tau)^{n-1}$. Consequently $X(i) = K\tau(1-\tau)^{n-1}$. Since there are n stations, $M = Kn\tau(1-\tau)^{n-1}$. We can conclude that stations access the channel equally.

As a result, the individual throughput of *station* i is $S_i = \frac{1}{n}S$, regardless of the station's data rate. This is validated by the simulations presented in the following sections.

III. Throughput Formulation for Mixed Data Rate

We first evaluate the throughput when different stations have different data rates but the same packet size. The protocol gives each station the same chance to transmit, and different data rates only affect average slot duration. Suppose there are n stations, D different data rates, $R^1 < \cdots < R^D$, with n^i stations in class i, i.e., with the same data rate R^i with corresponding slot durations T_s^i and T_c^i. For example, with eight stations $N = 8$ and four data rates $D = 4$ as in Figure 10, we get the following classification, arranged from the highest to the lowest duration:

$n^1 = 4$	T_s^1	$T_s(R^1)$	T_c^1	$T_c(R^1)$
$n^2 = 1$	T_s^2	$T_s(R^2)$	T_c^2	$T_c(R^2)$
$n^3 = 1$	T_s^3	$T_s(R^3)$	T_c^3	$T_c(R^3)$
$n^4 = 2$	T_s^4	$T_s(R^4)$	T_c^4	$T_c(R^4)$

The successful duration values can be evaluated by averaging the successful duration value of each station since only one station is involved in a successful duration. Since we know that each station has probability P_s/n of having a successful transmission, the new successful duration value $\bar{T}_s$ is

$$\begin{aligned} \bar{T}_s &= E[T_s^i] \\ &= \tfrac{P_s}{n} \textstyle\sum_{i=1}^{D} n^i T_s^i. \end{aligned} \tag{11}$$

When calculating the collision duration, we have to consider all the stations that are involved in the collision and how many times they are involved. A collision's duration value is determined by the station with the lowest data rate. The average collision duration $\bar{T}_c$ is given by

$$\begin{aligned} \bar{T}_c &= \textstyle\sum_{i=1}^{n-1} \sum_{j=1}^{D} \sum_{k=1}^{n^j} \begin{pmatrix} n - k - \sum_{l=1}^{j-1} n^l \\ i \end{pmatrix} \\ &\quad \times T_c^j \tau^{i+1} (1-\tau)^{n-1-i} \end{aligned} \tag{12}$$

This formulation considers the collision with the lowest data rate stations first. We then determine how many times the stations are involved in the collision with how many other stations. This process is repeated from the lowest data rate to the highest. In each iteration, lower data rates are excluded in the collision count.

Note that equation (12) is valid for all sets of data rate choices. For example, 802.11b has 4 and 802.11a has 8 data rate options. Furthermore, the equation is independent of the chosen Markov model. Any model that provides the transmission probability τ of a station could use the formula to obtain the collision duration values. The formula also works if the stations have different packet sizes. In that case the grouping into classes should be based on duration.

The throughput of a station is now given by

$$S_i = \frac{1}{n} \frac{P_s P(i)}{(1 - P_{tr})\sigma + \bar{T}_s + \bar{T}_c}. \tag{13}$$

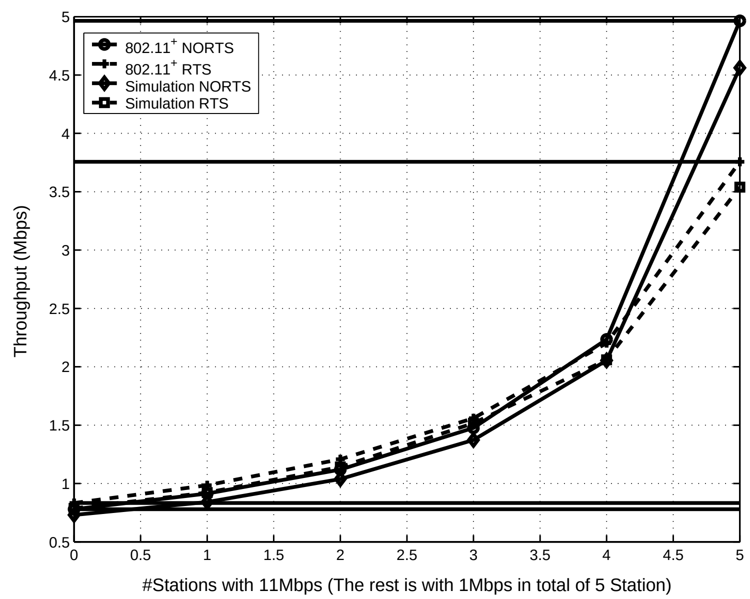

Fig. 11. Throughput verification for the mixed data rate formulation

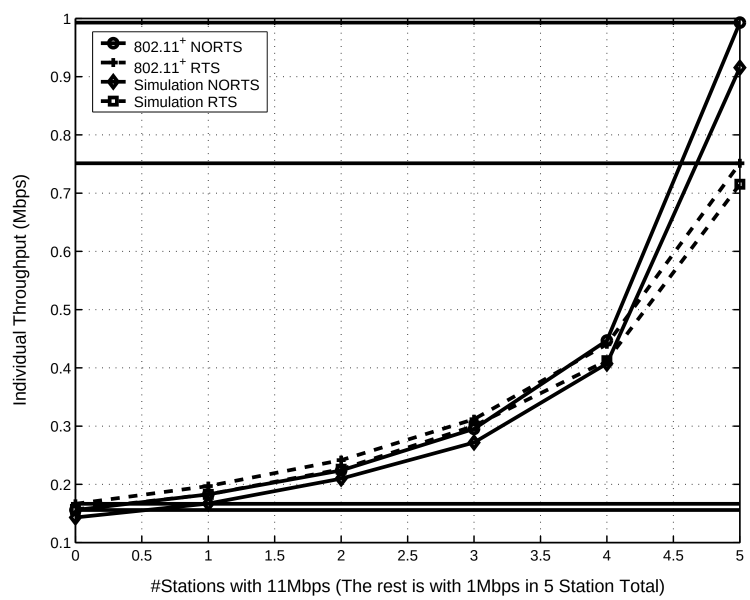

Fig. 12. Individual throughput verification for the mixed data rate formulation

Note that the individual throughput S_i is the *same* for all stations and total throughput S is nS_i if they have the same packet size and they equally access the channel.

We validate formula (12) with a simulation of 5 nodes and $D = [\ 1 \quad 2 \quad 5.5 \quad 11\ ]$Mbps. $E[P]$ is 1000 bytes and stations are saturated. Note that *RTS* and *CTS* packets are always sent at 1Mbps.

We start each station with 1Mbps and in each step we shift one of them to 11Mbps. At the fifth step we have 5 stations each with 11Mbps. We compare our results with an *OPNET* simulation.

Figures 11 and 12 show that the analytical formula (13) closely approximates the simulation-based performance. Solid lines represent the case when all stations have the same data rate. The individual throughput of the stations is also found to be equal in the simulation, and the throughput distribution among stations is verified to be equal. One individual realization is plotted in Figure 12.

As can be inferred from these plots, when there are 4 stations with 11Mbps and one station with 1Mbps, the throughput is almost half that when all are with 11Mbps. A lower data rate causes a considerable degradation for all stations.

IV. Optimization

A. Packet Size Adjustment

Let us revisit the individual station formula represented in equation (13). Optimal packet size for a given station can be found by taking the derivative of S_i with respect to $P(i)$, with duration values $T_s(R(i), P(i))$ and $T_c(R(i), P(i))$. We find that highest throughput is achieved when the stations with lower data rate are turned off. Although this achieves the maximum throughput, it is not fair, since the fair allocation gives *equal amount of channel usage*. This could be achieved by varying packet size with the data rate. As the data rate goes low, the stations can send lower sized packets preventing an unfair allocation to the high data rate stations.

Of course, this method can be viewed as increasing the packet size of the fast station with respect to slow station's packet size. However, there are some tradeoffs. Packet size has upper and lower limits; larger packets are more susceptible to errors; and, on the other hand, smaller packets can result in an under-utilized channel. This mechanism can be adaptively used depending on the packet size of the slow and fast stations. This method is simple and effective and can be used within the 802.11 standard.

We round down the duration values to the duration value of the highest data rate because the throughput increases monotonically with the increase in packet size (See Figure 13). Intuitively, the optimal packet size for the lower data rate station is the packet size that gives the same duration as the highest data rate station. If the highest data rate is R^h with packet size P_{R^h} then packet size of station with R^i data rate is given for IEEE 802.11b [8], [6] by

$$P_{R^i} = \lfloor \frac{R^i * P_{R^h}}{R^h} - 30 * \frac{(R^i - R^h)}{R^h} \rfloor. \tag{14}$$

As a result the data rate and packet size are related as follows:

$R(i)$ (Mbps)	$P(i)$ (bytes)
11	1000
5.5	515
2	206
1	118

This channel access mechanism equalizes channel usage. Stations can hear each other and if they hear a station with higher data rate then stations decrease their packet size to equalize their duration values with high data rate stations. (Increasing the data rate is not a solution since it is determined by the wireless channel in conformity with the physical layer standard.)

If P_{R^h} and $T_s(R^h, P_{R^h})$, $T_c(R^h, P_{R^h})$ represent the packet size and duration values of the station with the highest data rate then, then for all i,

$$\begin{aligned} T_s(R(i), P(i)) &= T_s(R^h, P_{R^h}) \ , \\ T_c(R(i), P(i)) &= T_c(R^h, P_{R^h}) \ . \end{aligned} \tag{15}$$

As a result the individual throughput formula equation (11) is modified as follows:

$$\bar{T}_s = P_s T_s(R^h, P_{R^h}), \tag{16}$$

$$\bar{T}_c = (P_{tr} - P_s) T_c(R^h, P_{R^h}), \tag{17}$$

$$S_i = \frac{1}{n} \frac{P_s P(i)}{(1 - P_{tr})\sigma + \bar{T}_s + \bar{T}_c}. \tag{18}$$

Figure 14 shows the improvement in throughput. Stations with 1 Mbps data rate decrease their packet size from

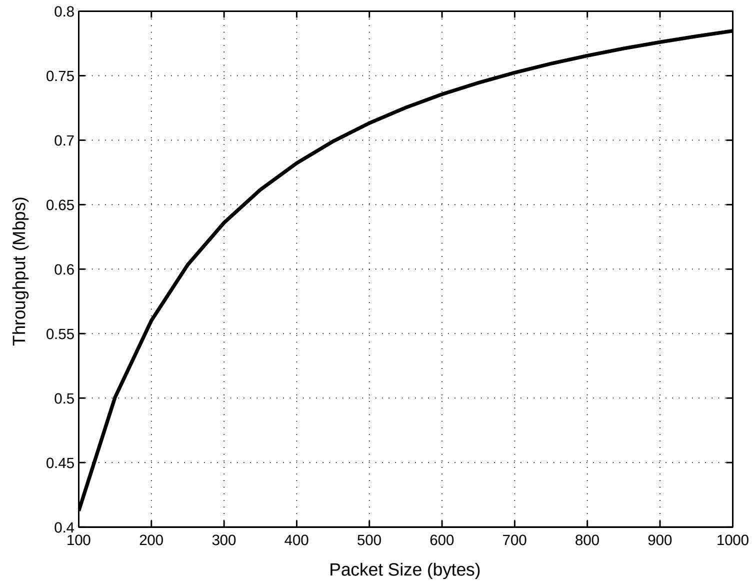

Fig. 13. Throughput with packet size

1000 bytes to 118 bytes if a 11 Mbps station is present.

The advantage is in total throughput. On the other hand, in terms of individual throughput, high data rate stations are better off but the low data rate stations are worse off, as seen in Figure 15.

Case Study:

We apply this mechanism to our original scenario in which there are five 11Mbps stations except for one station which changes its data rate during the simulation. Figure 16 shows the transmitted data. The upper figure shows the transmitted data for fast and slow stations; and the lower figure shows the channel activity of the slow station.

Figure 17 shows the total throughput compared with the situation before optimization, i.e., without the mechanism. As expected, the improvement in total throughput decreases as the data rate of the slow station increases.

Note that a station's throughput in *OPNET* is the *successfully received bits* per unit time; let's call this the "received throughput" (S^r). Keeping this definition in mind, we see in Figure 18 that, with optimization, the slow station has a higher received throughput; hence it is receiving more packets. This is explained as follows. The optimization algorithm suppresses the transmission of the slow station and give more time to the fast stations, which therefore send more packets to the slow station. On the other hand, the slow station sends fewer packets to

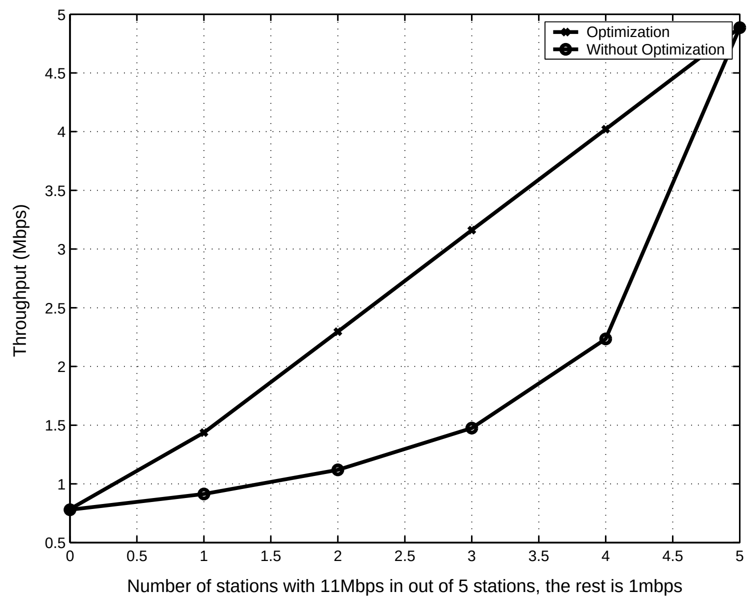

Fig. 14. Throughput after optimization (w/o RTS/CTS)

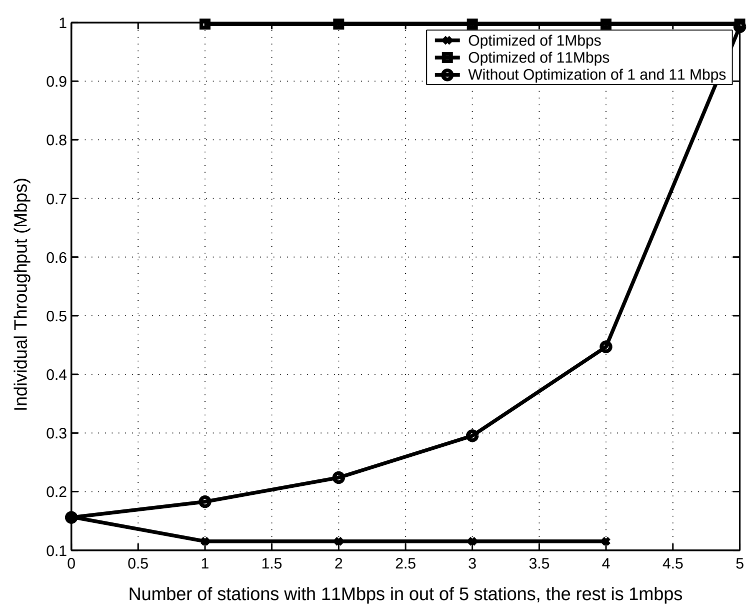

Fig. 15. Individual throughput after optimization (w/o RTS/CTS)

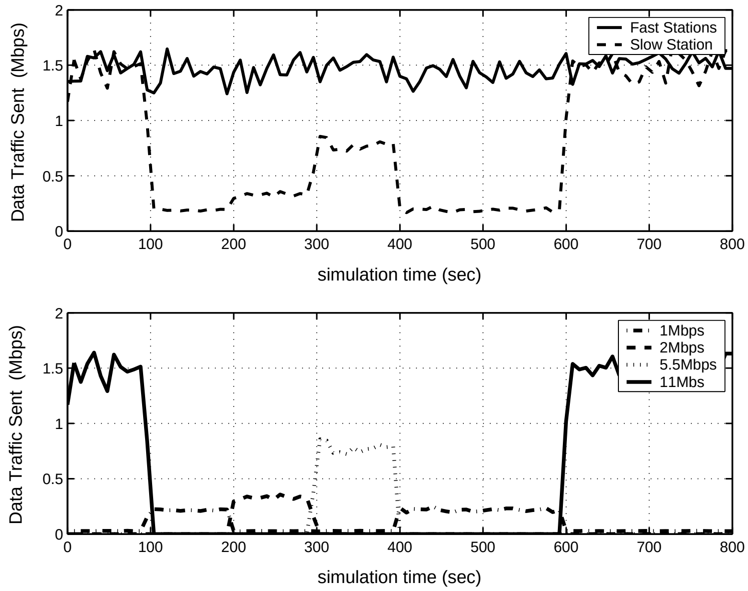

Fig. 16. Data traffic sent

the fast stations.

If we define throughput as *successfully transmitted bits over time* and call it "transmitted throughput" (S^t) as in our Markov model analysis, we can extract these values from the received throughput. We have 4 fast stations and 1 slow station. So the slow station's received throughput comes from 4 fast stations, each contributing one fourth of the received traffic. On the other hand, a fast station's throughput comes from 3 fast stations and 1 slow station. We can conclude that transmitted and received throughput are related by

$$S^t_{Fast} = S^r_{Slow}$$

$$S^t_{Slow} = |4S^r_{Fast} - 3S^r_{Slow}|$$

and the result is depicted in Figure 19.

If we consider throughput as the number of packets that has been sent correctly we infer that slow station should have the lower throughput as compared to case prior to optimization. This is because the slow station sends fewer bits which is shown in Figure 15. We make the same conclusion by looking at the medium access delay, which is the waiting time of the packets in the MAC layer queue. Figure 20 shows the medium access delay for the stations. As expected, the slow station has the highest medium access delay and fast stations have the lowest delay. Figure 21

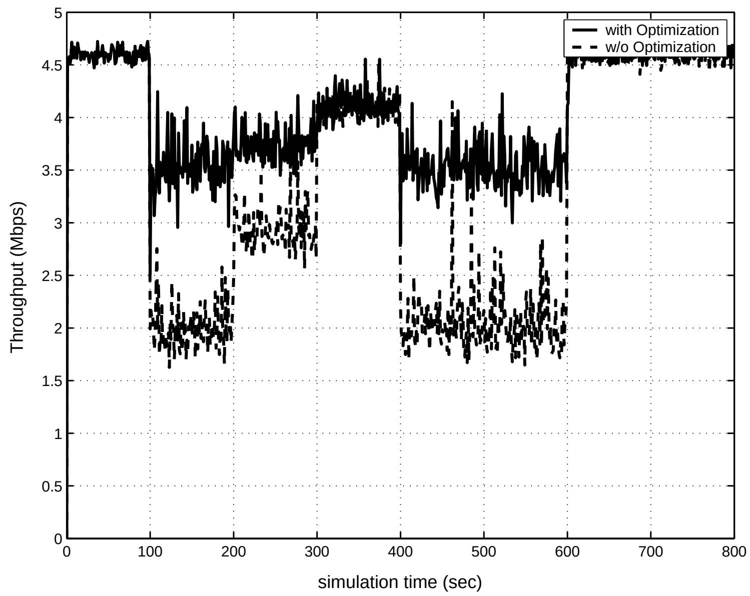

Fig. 17. Total throughput

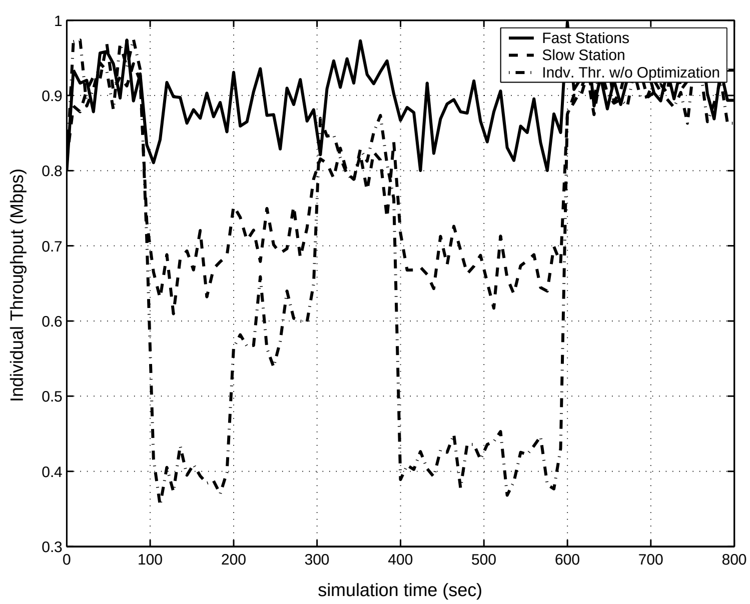

Fig. 18. Individual throughput (Received throughput (S_i^r))

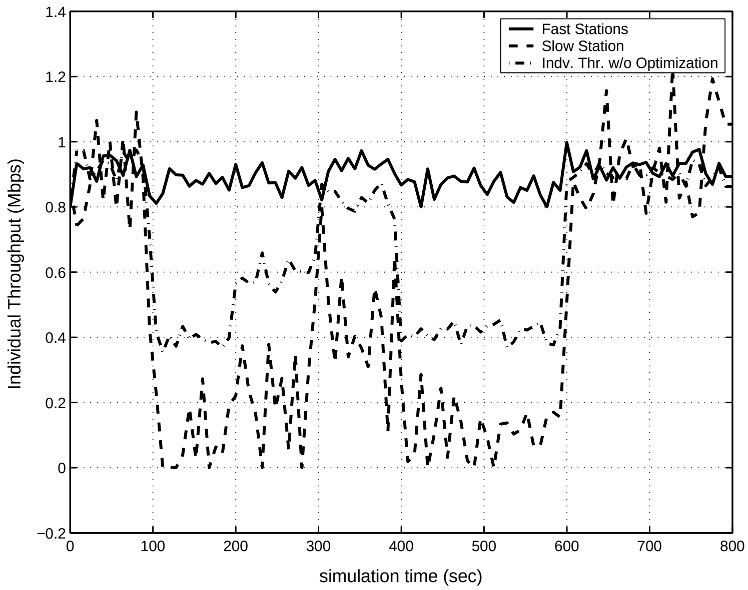

Fig. 19. Individual throughput (Transmitted throughput (S_i^t))

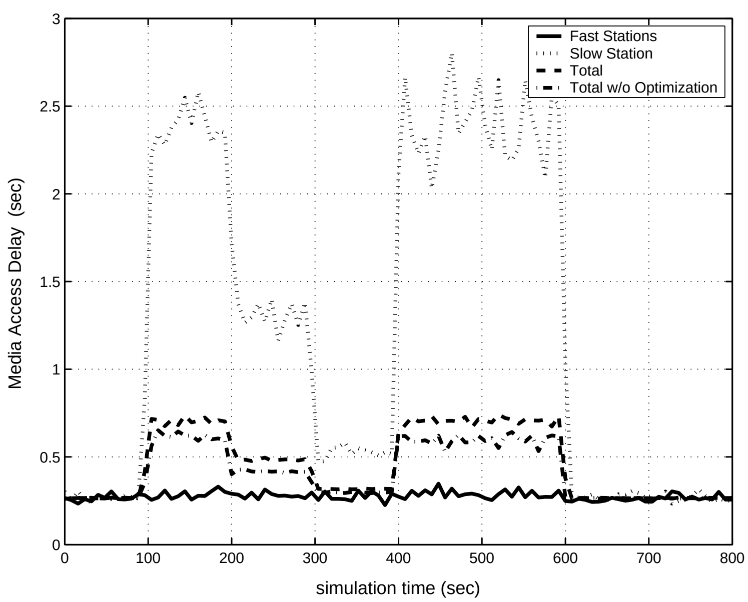

Fig. 20. Medium Access Delay

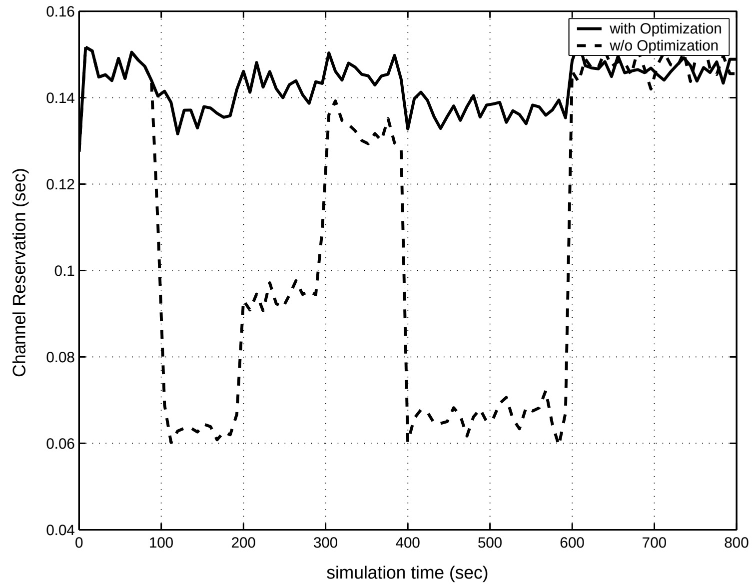

Fig. 21. Channel reservation

shows that the channel reservation, which is a metric that measures channel utilization, has increased considerably as a result of the optimization.

B. Adaptive Packet Size Adjustment

We can also formulate a scenario for packet size aggregation, in which a station is prioritized with permission to send larger sized packets compared to others.

Suppose there are $(n-1)$ stations and an access point in a network. Let's say that each station performs a voice session with a node outside the network [28], [29]. Let's mark *station 1* as the *access point*. We suppose that without optimization packet sizes are equal and given by

$$E[P] = P(1) = P(i) \ for \ i \in \ [2, n].$$

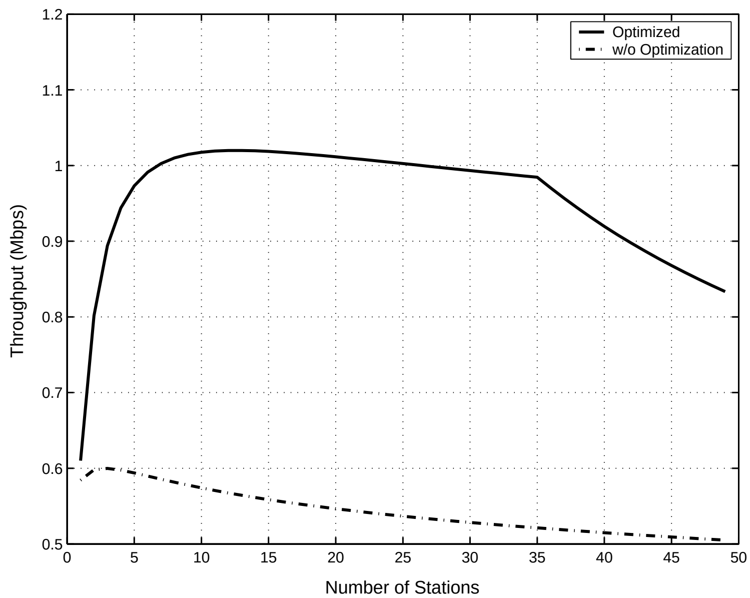

Fig. 22. Total throughput (w/o RTS/CTS)

With packet size adjustment, the packet sizes are

$$P(1) = \min((n-1)(P(2)+I), 2024)$$

$$P(2) = P(i) \; for \; i \in \; [2, n]$$

$$E[P] = \frac{1}{n} \sum_i^n P(i)$$

where I is the number of identifier bytes and 2024 bytes is the maximum allowed payload. The identifier is used by the stations to extract their payload from the access point's transmission. We considered 20ms segment at 8kbps which corresponds to 64 bytes $MSDU$ ($P(2)$) for a single voice session and each station has a packet to transmit all the time with 11 Mbps data rate. In addition, the access point does not expect an ACK packet since it transmits in multicast.

Figure 22 shows the total throughput S_T of the system found with the formula explained in Section III:

$$S_T = \frac{P_s E[P]}{(1-P_{tr})\sigma + \bar{T}_s + \bar{T}_c}. \tag{19}$$

One can see the substantial throughput increase from the figure. As can be inferred, the packet size of the

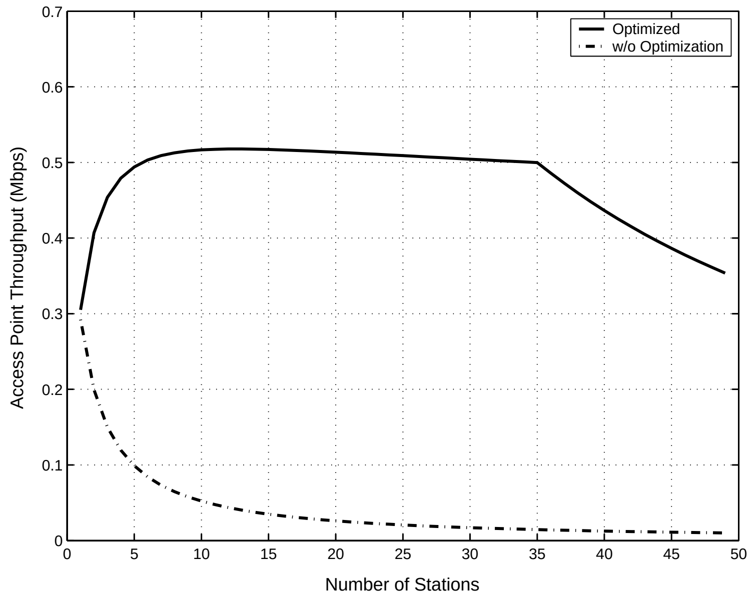

Fig. 23. Access point throughput (w/o RTS/CTS)

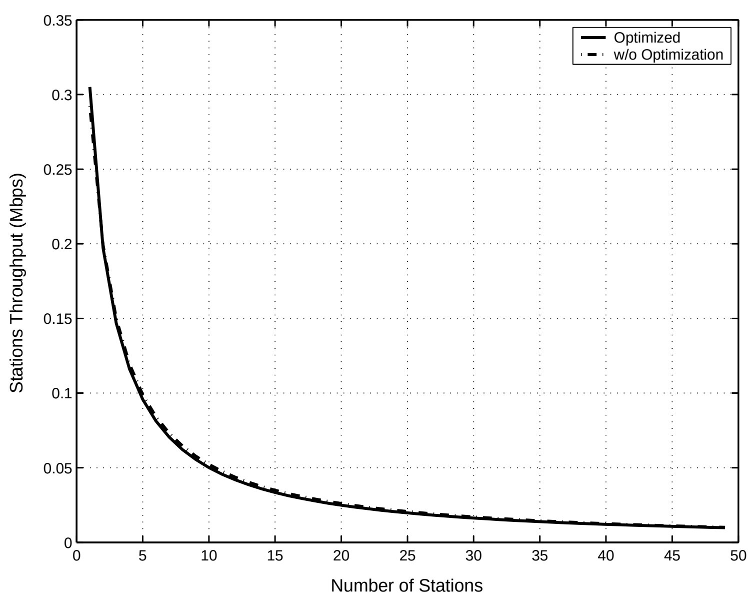

Fig. 24. Station throughput (w/o RTS/CTS)

access point increases with the number of stations, which increases the throughput. On the other hand, increase in the number of stations decreases the throughput because of the increased probability of collision. Eventually the maximum allowed packet size is reached, and the throughput starts to decrease.

Figure 23 shows the access point throughput S_A found with the formula

$$S_A = \frac{1}{n} \frac{P_s P(1)}{(1 - P_{tr})\sigma + \bar{T}_s + \bar{T}_c}. \tag{20}$$

Sending packets in bursts increases the throughput of the access point. This throughput calculation considers the successfully transmitted bits but not the received bits. As a result, stations receive packets at higher rates and the station's transmitted throughput S_S is found with the formula

$$S_S = \frac{1}{n} \frac{P_s P(2)}{(1 - P_{tr})\sigma + \bar{T}_s + \bar{T}_c}. \tag{21}$$

Figure 24 depicts the comparison. As expected, a station's throughput S_S with this algorithm should be lower than that without the algorithm. This is because higher packet size of the access point increases the average duration values. One can easily see that the following equation holds

$$S_T = S_A + (n-1)S_S.$$

C. Packet Aggregation

From Figure 13 we can infer that sending larger packets increases the throughput. We can conclude that each station should aggregate the small sized packets into a big one and then send it [30]. In a scenario in which only the access point aggregates the packets into a 2024 bytes size packets ($P(1)$), we obtain a performance increase which gradually decreases as a new station joins. Figure 25 shows the total throughput and Figures 26 and 27 show the access point throughput and stations throughput, respectively. In frame aggregation the traffic could have one destination or multiple destinations. If multicast, there is no need for an ACK packet. Otherwise, with only one destination, one ACK is needed and, for multiple destinations, each station may send an ACK packet one by one with respect to their order in the received packet.

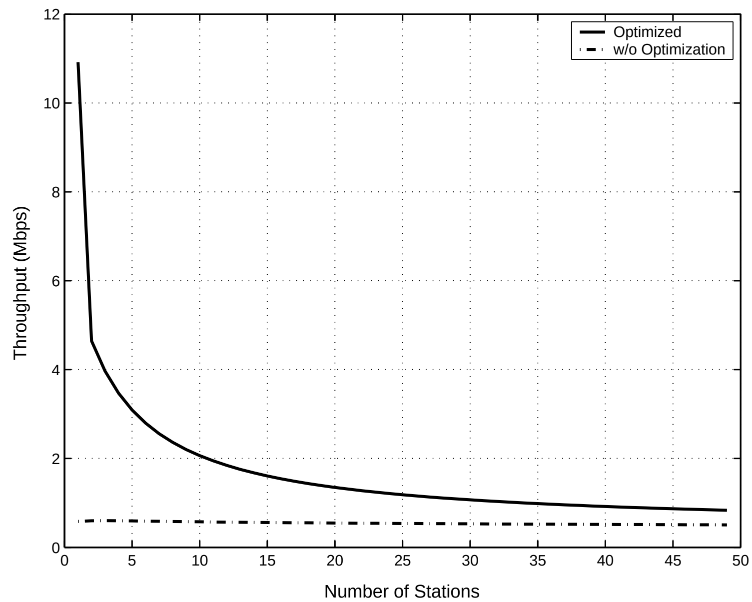

Fig. 25. Total throughput for frame aggregation (w/o RTS/CTS)

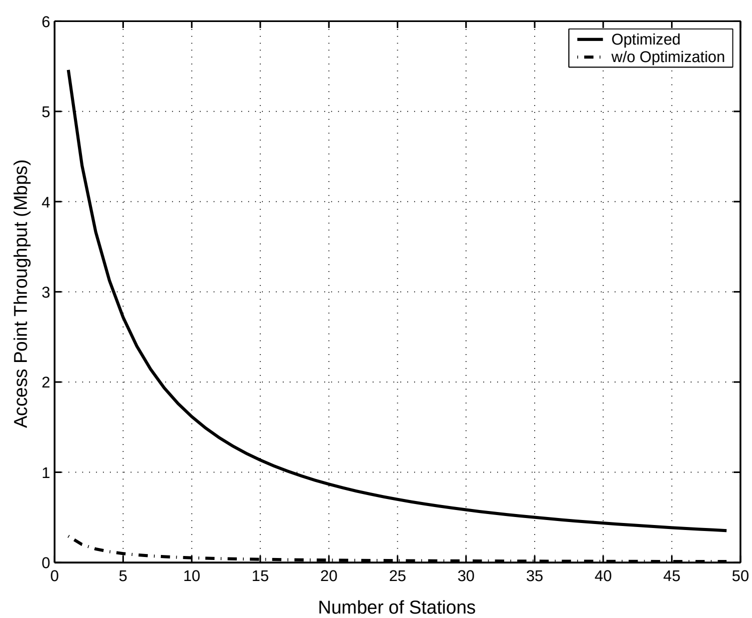

Fig. 26. Access point throughput for frame aggregation (w/o RTS/CTS)

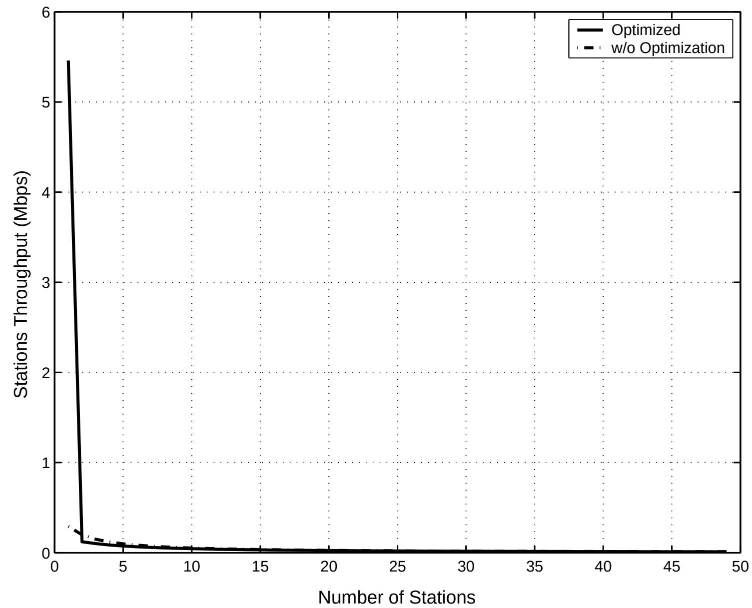

Fig. 27. Station throughput for frame aggregation (w/o RTS/CTS)

V. Conclusion

The *CSMA/CA* scheme does not provide a fair scheduling in terms of giving equal channel usage to each station, but it does provide equal chance of channel access. We analyzed the case when there is a low-rate station in the network and concluded that throughput could be increased by turning that station off. A fair allocation, however, should give equal amounts of channel usage per station. To achieve this, we introduced a mechanism to adapt the packet size with respect to the data rate. As a result, the slow station decreases its packet size to occupy the same amount of channel usage as the fast station. This is another way of saying to increase the packet size of the fast station to have the same channel usage as the slow station. One or the other rule can be adopted.

This scheme is very simple and can be adopted without a change in IEEE 802.11 standard. This makes this scheme novel compared to existing fairness algorithms.

References

[1] I. AAd, C. Castelluccia. *Differentiation mechanism for IEEE 802.11.* IEEE INFOCOM, 2001.

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

[3] IEEE 802.11b. *Wireless LAN, Medium Access Control (MAC) and Physical Layer (PHY) Specifications: High-Speed Physical Layer Extension in the 2.4GHz Band.* Supplement to IEEE 802.11 Standard, September 1999.

[4] IEEE 802.11a. *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications: High-Speed Physical Layer Extension in the 5 GHz Band.* Supplement to IEEE 802.11 Standard, September 1999.

[5] IEEE 802.11g. *Further Higher-Speed Physical Layer Extension in the 2.4GHz Band.* Supplement to IEEE 802.11 Standard, 2003.

[6] G. Bianchi. *Performance Analysis of the IEEE 802.11 Distributed Coordination Function.* IEEE Journal on Selected Areas in Communications, vol.18, March 2000.

[7] A. Kumar, E. Altman, D. Miorandi, M. Goyal, *New Insights from a Fixed Point Analysis of Single Cell IEEE 802.11 WLANs.* IEEE INFOCOM 2005.

[8] M. Ergen. *I-WLAN: Intelligent Wireless Local Area Networking.* PhD Dissertation, UC Berkeley, December 2004.

[9] M. Heusse, F. Rousseau, G. Berger-Sabbatel, A. Duda. *Performance Anomaly of 802.11b.* Proceedings of IEEE INFOCOM 2003.

[10] G. R. Cantieni, Q. Ni, C. Barakat, T. Turletti. *Performance Analysis under Finite Load and Improvements for Multirate 802.11.* preprint submitted to Elsevier Science.

[11] R. Bruno, M. Conti, E. Gregori, R. Fantacci. *Throughput vs. Temporal Fair MAC protocols in Multi-Rate WLANs: Analyses and Performance Evaluation.* IEEE VTC 2004-Spring.

[12] M. Ergen, P. Varaiya. *Individual Throughput with Different Data Rates from Markov Model of IEEE 802.11.* Proceedings of IEEE GLOBECOM-CAMAD, December 2004.

[13] A. Kamerman, L. Montean. *WaveLAN-II: A High-Performance Wireless LAN for the Unlicensed Band.* Bell Labs Technical Journal, pp. 118-133, Summer 1997.

[14] G. Holland, N. Vaidya, P. Bahl. *A Rate-Adaptive MAC Protocol for Multi-hop Wireless Networks.* MOBICOM 2001.

[15] D. Qiao, S. Choi, K. Shin. *Goodput Analysis and Link Adaptation for IEEE 802.11a Wireless LANs. IEEE Commun. Lett.,* vol. 7, no. 2, pp. 70-72, Feb. 2003.

[16] Z. Kong, D. H. K. Tsang, B. Bensau, D. Gao. *Performance Analysis of IEEE 802.11e Contention-Based Channel Access.* IEEE Journal on Selected Areas in Communications, vol.22, December 2004.

[17] B. Sadeghi, V. Kanodia, A. Sabhanarwal, E. Knightly. *Opportunistic Media Access for Multirate Ad Hoc Networks.* MOBICOM 2002.

[18] B. Bensaou, Y. Wang, and C. Ko. *Fair Medium Access in 802.11 Based Wireless Ad-Hoc Networks.* MobiHoc 2000.

[19] F. Cali, M. Conti, E. Gregori. *Dynamic Tuning of the IEEE 802.11 Protocol to Achieve a Theorethical Throughput Limit. IEEE/ACM Trans. Networking,* vol. 1., no.1. pp. 10-31, March 2002.

[20] M. Malli, Q. Ni, T. Turletti, C. Barakat. *Adaptive Fair Channel Allocation for QoS Enhancement in IEEE 802.11 Wireless LANs.* IEEE ICC 2004.

[21] H. S. Chahalaya, S. Gupa, *Throughput and fairness properties of asynchronous data transfer methods in the IEEE 802.11 MAC protocol.* Personal Indoor Mobile and Radio communication conference, pp. 613-617, 1995.

[22] K. C. Chen. *Medium access control of wireless LANs for mobile computing. IEEE Network,* vol. 8, no. 5, pp. 50-63, September 1994.

[23] H. S. Chhaya, S. Gupta. *Peformance modeling of asynchronous data transfer methods of IEEE 802.11 MAC protocol. Wireless networks vol. 3, 1997.*

[24] S. Lu, V. Bharghavan, R. Srikant. *Fair Scheduling in Wireless Packet Networks. IEEE Commun. Lett.,* vol. 7, no. 2, pp. 70-72, Feb 2003.

[25] N. Vaidya, P. Bahl, S. Gupta. *Distributed Fair Scheduling in a Wireless LAN.* MOBICOM 2000, Aug. 2000.

[26] Y. Xiao, *A Simple and Effective Priority Scheme for IEEE 802.11. IEEE/ACM Trans. Networking,* vol. 7, no. 4, pp. 473-489, Aug. 1999.

[27] M. Ergen, P. Varaiya. *Admission Control and Throughput Analysis in IEEE 802.11. Springer Mobile Networks and Applications,* vol. 10, no. 5., pp. 705-706, October 2005.

[28] J. Liesenborgs. Voice over IP in networked virtual elements. Ph.D dissertation, University of Maastricht, 2000.

[29] S. Garp, M. Kappes. *Can I add a VoIP call?* Proceedings of IEEE ICC 2003, May 2003.

[30] H. Tounsi, L. Toutain, F.Kamaoun. *Small packets aggregation in an IP domain.* Proceedings of Sixth IEEE Symposium on Computers and Commuincations, July 2001.