COMMUNICATIONS IN INFORMATION AND SYSTEMS
Vol. 3, No. 4, pp. 279-304, September 2004

003

A PERFORMANCE ANALYSIS OF THE 802.11 WIRELESS LAN MEDIUM ACCESS CONTROL*

NITIN GUPTA† AND P. R. KUMAR‡

Dedicated to Sanjoy Mitter on the occasion of his 70th birthday.

Abstract. We study the performance of the IEEE 802.11 protocol. We present an extension of a methodology for the collocated one-hop case which allows the incorporation of channel errors. The results closely agree with simulation results. A delay analysis is also presented. We also present an extension of this methodology to the multi-hop case with non-collocated nodes. The approach uses specific topology dependent relations. Specific results are presented for the ring and mesh topologies, and compared against simulation results.

1. Introduction. Wireless networking technologies are increasingly becoming widespread. Various wireless communication standards have evolved which try to provide protocols and standards for medium access control in the shared wireless medium. The IEEE 802.11 protocol[3], Bluetooth[7], HomeRF[9] (now disbanded) and the HiperLAN[8] are examples of such standards. The IEEE 802.11 protocol is the most widely used.

The IEEE 802.11 standard defines two layers. The first layer is the Physical layer (PHY), which specifies the modulation scheme used and signaling characteristics for the transmission through radio frequencies. The second layer is the media access control (MAC) layer. This layer determines how the medium is used. This chapter provides an overview of the medium access control mechanism, specifically the Distributed Coordination function, DCF, in the IEEE 802.11 protocol.

The description is followed by performance analysis of DCF for both single hop and multi-hop wireless networks. Bianchi in [1] has analyzed IEEE 802.11 DCF for the single hop or the collocated case. The goodput analysis for the collocated case presented in Section 3.1, extends Bianchi's work by taking into account wireless channel errors and retry counts. A new delay analysis for the collocated case is also presented. Analyzing 802.11 protocol in multi-hop scenarios poses many new challenges. Section 3.2 presents a methodology for performance analysis of 802.11 DCF in multi-hop networks. The application of the methodology to a ring topology is also presented. The numerical results show good agreement with ns-2 simulation

*Accepted for publication on July 2, 2004. Please address all correspondence to the second author at the address shown below. This material is based upon work partially supported by AFOSR under Contract No. F49620-02-1-0217, USARO under Contract Nos. DAAD19-00-1-0466 and DAAD19-01010-465, AFOSR under Contract No. F49620-02-1-0325, and NSF under Contract Nos. NSF ANI 02-21357 and CCR-0325716.

†Department of Computer Science, University of Illinois at Urbana Champaign, Urbana, IL 61801.

‡University of Illinois, Coordinated Science Laboratory, 1308 West Main St., Urbana, IL 61801-2307, USA. Email: prkumar@uiuc.edu. Web: http://black.csl.uiuc.edu/∼prkumar.

results.

2. The IEEE 802.11 Medium Access Control protocol: A Description. The IEEE 802.11 protocol defines a basic service set (BSS). A BSS is a set of stations that communicate with one another. When all of the stations in the BSS are mobile stations and there is no connection to a wired network, the BSS is called an *independent BSS* (IBSS), or what is commonly called the ad hoc mode. When a BSS includes an access point (AP), the BSS is called the *infrastructure BSS*.

The 802.11 specification supports two fundamentally different MAC schemes, namely the *distributed coordination function DCF*, and the *point coordination function PCF*. DCF is designed to support asynchronous data transport where all users have an equal chance of accessing the network. PCF is designed for transmission of delay sensitive data. The PCF is built on top of the DCF, and is used only on infrastructure networks. The PCF mode is not widely implemented and so we will not discuss it any more.

In this section we will concentrate on DCF and the ad hoc mode of operation with no access point. The DCF protocol can be described as carrier sense multiple access with collision avoidance (CSMA/CA). There are two access methods that are used under DCF, namely the *basic access method* and the *RTS/CTS access method*. This section describes both methods. We also describe the backoff procedure, timing intervals, and retry counts, that are defined by the protocol.

2.1. Frame types. DCF defines four types of MAC layer frames, namely RTS (request to send), CTS (clear to send), DATA and ACK (acknowledgment).

2.2. Timing Intervals. There are five timing intervals:

1. The short interframe space (SIFS).
2. The slot time, represented by σ.
3. The priority interframe space (PIFS).
4. The distributed interframe space (DIFS).
5. The extended interframe space (EIFS).

The timing intervals are determined by the physical layer. The physical layer standard defines *frequency hopping spread spectrum FHSS* and *direct sequenced spread spectrum DSSS* protocols. For the Frequency hopping spread spectrum (FHSS) physical layer, SIFS is 28 μs and slot time is 50 μs and for DSSS, SIFS is 10 μs and slot time is 20 μs. The PIFS is equal to SIFS plus one slot time. The DIFS is equal to the SIFS plus two slot times. The EIFS is much larger than any of the other intervals. It is used when a frame that contains errors is received by the MAC, allowing the possibility for the MAC frame exchanges to complete correctly before another transmission is allowed.

2.3. Virtual Carrier Sensing. DCF uses a virtual carrier sensing mechanism in addition to the physical carrier sensing. This is accomplished by a variable called the *network allocation vector NAV*. A node regards the channel as busy if the physical carrier sense indicates the medium is busy, or if the NAV is set to a non-zero value. This mechanism is known as *virtual carrier sensing*, and is used by both the basic access method and the RTS/CTS access method. In the following paragraphs by "sensing the channel" we will mean both physical carrier sensing as well as a non-zero NAV value.

2.4. The DCF Basic Access Method. This method uses only DATA frames and ACK frames.

The station which wishes to send a data frame first senses the channel, and may proceed with transmission of a DATA frame if the medium is sensed to be idle for an interval larger than DIFS. When the DATA frame is transmitted, all other stations hearing the frame set their network allocation vector, NAV, based on the duration field value in the data frame received. The duration field in the data frame includes the SIFS and the ACK frame transmission times following the DATA frame. By using this virtual carrier sensing mechanism, all stations that hear the DATA frame will refrain from transmitting during the ACK frame transmission, thus reducing frame collisions.

If the medium is busy, the transmitting station will wait until the end of the current transmission. It will then further wait for an additional DIFS time period, and then the station defers for a random backoff period before transmitting. The Backoff Procedure is described in Section 2.7.

Upon having received a DATA frame correctly, the destination station waits for a SIFS interval immediately following the reception of the DATA frame, and then transmits an ACK frame back to the source station, indicating that the DATA frame has been received correctly. We illustrate the basic access method in Figure 1.

2.5. The hidden node problem. Consider the three stations illustrated in Figure 2. Their transmission ranges are shown. A and C cannot hear each other, and if they both transmit at the same time to B, their DATA frames will collide, resulting in frame loss. This is known as the *hidden node problem.* Even virtual carrier sensing cannot avoid this problem. DCF addresses this problem through the RTS/CTS access method which uses small RTS and CTS frames prior to sending the large DATA frames.

2.6. The DCF RTS/CTS Access Method. This method uses a four-phase RTS-CTS-DATA-ACK handshake. Figures 3 and 4 illustrate the RTS/CTS access method.

The station that wishes to send a DATA frame first senses the channel. If the channel remains idle for a DIFS interval, then it sends an RTS frame. Otherwise, it

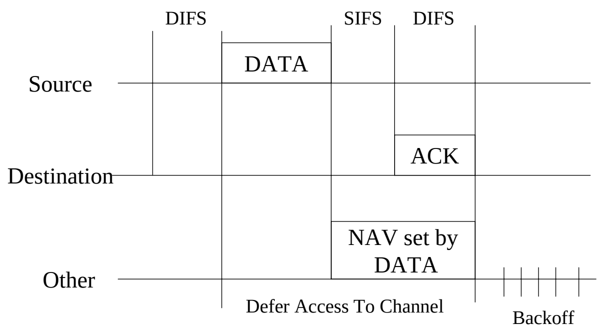

FIG. 1. *The Basic Access Method.*

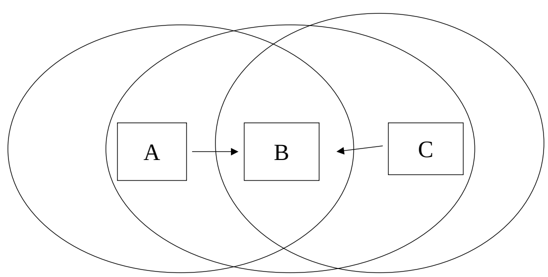

FIG. 2. *The hidden node problem.*

triggers the backoff algorithm after waiting till the end of current transmission and a further DIFS period.

When the destination receives the RTS, it transmits a CTS after a SIFS interval. The source station is then allowed to transmit its DATA frame after a time interval corresponding to a SIFS, after successful reception of the CTS frame. All other stations which hear the RTS or the CTS frame update their NAV by the duration fields in the RTS and CTS frames. The duration field in the RTS frame is set to the sum of the time durations for transmission of CTS, DATA, and ACK frames, plus three SIFS intervals. The duration field in the CTS frame is set to the sum of the

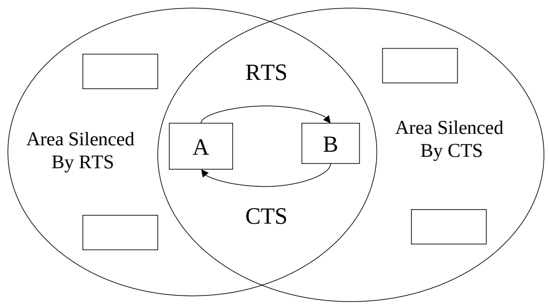

Fig. 3. *The RTS/CTS exchange silences two neighborhoods;those of A and B.*

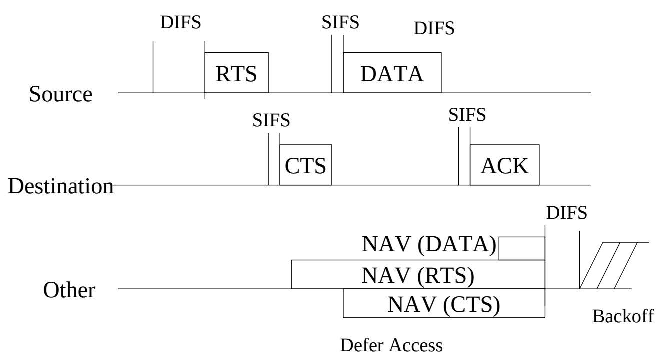

Fig. 4. *The RTS/CTS Access Method.*

time durations for transmission of DATA, and ACK frames, plus two SIFS intervals. Thus, virtual carrier sensing is used to silence the neighbors of the source A as well as the destination station B, as shown in Figure 3.

Using this four phase handshake, DATA-DATA frame collisions are avoided even in the presence of hidden terminals. RTS-RTS frame collisions are shorter, and result in lesser bandwidth loss than larger DATA-DATA frame collisions.

2.7. Backoff procedure. Every station has a *backoff counter* and a *backoff stage*. The backoff counter value is initially chosen as described below. The backoff procedure selects a random number of time slots between 0 and the *contention window CW*, according to the following equation:

BackoffCounter = INT(CW * Random()) * Slot time).

CW is an integer between CW_{min} and CW_{max}, typical values being 31 and 1023, respectively. Random() is a random number between 0 and 1. Slot time is fixed for a given physical transmission scheme.

The backoff stage is initially 0. The backoff counter is decreased by 1 for every slot, as long as the channel is sensed idle in that slot. If there are transmissions by other stations during the slot, then the station will freeze its backoff counter and will resume the count where it left off, after the other transmitting station has completed its frame transmission plus an additional DIFS interval.

Transmission commences when the counter reaches zero. If the transmission is successful, then the backoff counter is again chosen as described above, and the backoff stage is reset to 0. Upon an unsuccessful transmission, the backoff stage is increased by 1, and the backoff counter is chosen again between 0 and CW for the new backoff stage. The CW value for the next backoff stage is doubled, until it reaches CW_{max}, after which it becomes constant. Thus, typically, we have CW values as 31, 63, 127 for backoff stages 0,1,2 respectively, and so on.

2.8. Retry count. The IEEE 802.11 standard defines a *short Retry count*, which is the maximum number of RTS transmission attempts that will be made before a frame is discarded. Typically it is set to 7.

3. Performance analysis. This section presents analytical methods for performance analysis of IEEE 802.11 wireless LANs. The analysis models the exponential backoff algorithm and the virtual carrier sensing mechanism of the 802.11 MAC layer. The performance metric chosen is goodput attainable by each node. A brief delay analysis is also provided for a single hop 802.11 LAN.

Canti et al. in [2] provide an approximate analytical model for computing throughput in a 802.11 wireless network. The analysis presented assumes an approximate backoff algorithm instead of modeling the actual backoff algorithm. The assumption made is that backoff time for a station is geometrically distributed with a probability 1/(E[B]+1), where E[B] is the average value of a backoff period.

G. Bianchi in [1] models the DCF algorithm for single hop wireless networks. The analysis models 802.11 DCF using Markov Chains. The analysis does not take into account packet losses due to bit errors introduced by the channel, and retry counts defined by the protocol. The analysis for the single hop case presented in this section is an extension of Bianchi's work to account for wireless bit errors and also models the short retry count. The analysis can be further refined by introducing more dummy states into the Markov Chain as described below. A delay analysis is also presented based on the refined Markov Chain.

Modelling the 802.11 protocol for multi-hop scenarios is relatively complicated because of hidden terminals. Section 3.2 devoted to multi-hop scenarios explains the difficulty in analyzing multi-hop 802.11 wireless LANs. It provides approaches for approximate throughput computation in particular multi-hop scenarios. As an example, the analysis for a Ring topology is provided. The numerical results show good agreement with ns-2 simulation results. Actual experimentation is however needed to validate the approach.

3.1. Performance analysis of the single hop 802.11 wireless LAN. In this section, we model a wireless LAN where the nodes are all collocated, and so frames need only traverse one hop before reaching their destination.

Model Assumptions

1. There are no hidden stations, and all the stations can hear each other's transmissions.
2. The network consists of finite number n of contending stations.
3. The cyclic redundancy check CRC (included in every frame header) is perfect and erroneous packets are always detected. (The CRC is meant to detect if frames have been corrupted in transit.)
4. When a packet is corrupted by the wireless medium, all stations observe it as corrupted.
5. Every station always has a packet to transmit. So we are considering a saturated condition when the network is highly loaded.
6. The major assumption of the model is that the collision probability of a transmitted frame is constant and independent of the number of retransmissions attempts on this frame. This assumption from Bianchi's model [1] is very useful for simplifying the analysis.
7. The wireless channel model is assumed to be a two state Markov Chain with alternating good and bad states, as shown in Figure 5. The durations of the good states and bad states are assumed to be exponential random variables with means λ_g^{-1} and λ_b^{-1}, respectively.

3.1.1. Goodput analysis. The analysis methodology consists of two phases. We begin by assuming a fixed and independent probability τ for a station to be transmitting an RTS frame in a slot. As a function of τ, we then compute the goodput of the system as a whole by computing the probabilities and durations of all the possible events that the wireless channel undergoes, and in particular the probability p that an RTS is unsuccessful given that it has been attempted. This is a global analysis which considers all the n stations. In the second phase, we concentrate on a single

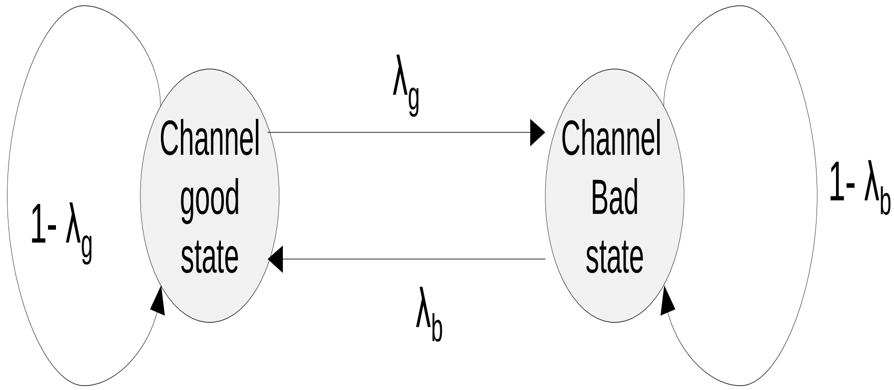

FIG. 5. *Model of wireless channel.*

station and model its backoff algorithm to compute τ as a function of p which is the probability that an RTS is unsuccessful given that it has been attempted. This is a local analysis. Therefore, we compute τ in the second phase, assuming p which was computed in the first phase, as a function of τ. Finally, we obtain τ as a fixed point of these two phases, completing the analysis. The analysis methodology is the same as used by Bianchi in [1]. The one new feature is that we model channel errors as above.

Global Analysis: Goodput Computation

Each node listens to the same channel in the collocated case. The wireless channel at any time is undergoing one of the following seven events: Successful packet transmission by one of the n nodes, RTS-RTS frame collision of two or more stations transmitting RTS frames in the same slot, RTS frame corruption, CTS frame corruption, DATA frame corruption, ACK frame corruption, and idle in which case no node is transmitting. There can be only RTS-RTS collisions in the collocated case as all nodes can listen to all other nodes. Figure 6 shows each of the seven events.

In the first phase, the key idea is to compute the probabilities for each of the seven events, and then compute the goodput. It is possible to compute probabilities for each of the events above in terms of τ, λ_g, and λ_b. The probabilities and durations for the seven cases are provided in Table 1. The quantity δ is used to denote the propagation delay. We now elaborate on some of the details of these computations.

We will observe an idle slot if none of the nodes transmits a RTS frame. Therefore,

$$p_1 = (1 - \tau)^n.$$

There will be an RTS-RTS collision if more than one station transmits a RTS in the same slot. So,

$$p_2 = 1 - (1 - \tau)^n - n\tau(1 - \tau)^{n-1}.$$

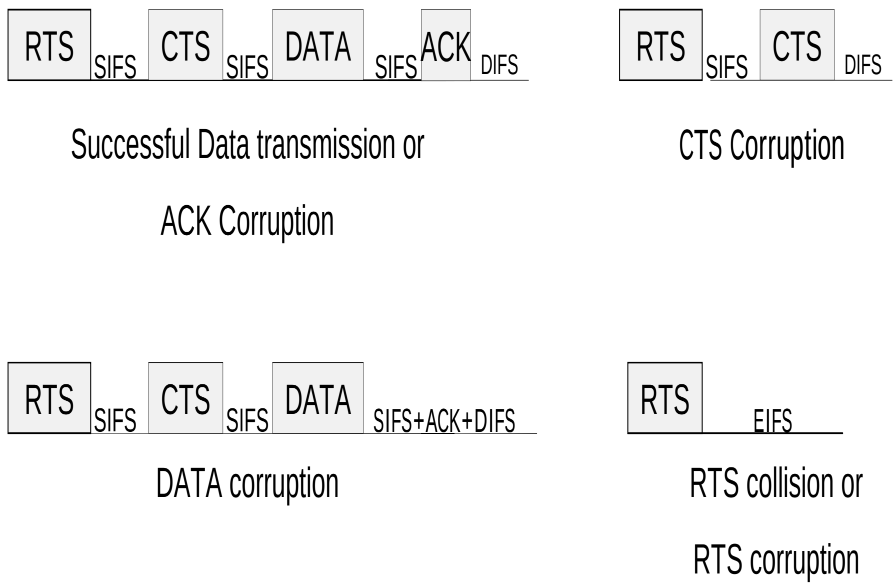

Fig. 6. *Events in the wireless channel*

The probability for RTS corruption given that there is no RTS collision is calculated as follows:

Pr(RTS gets corrupted | Exactly 1 RTS gets transmitted)

= 1−Pr(Chosen RTS falls entirely within a good state duration)

= 1−Pr(Beginning instant of RTS falls within a good state)

Pr(Good state duration > RTS+δ)

$= 1 - \frac{\lambda_b}{\lambda_g+\lambda_b} e^{-\lambda_g(RTS+\delta)}$.

Therefore we get

$$p_3 = n\tau(1-\tau)^{n-1}(1 - \frac{\lambda_b}{\lambda_g+\lambda_b} e^{-\lambda_g(RTS+\delta)}).$$

The computation of the CTS frame corruption probability, given that some RTS gets through successfully, is as follows.

$p_4 :=$ F(τ,n) $= n\tau(1-\tau)^{n-1}$ Pr(RTS is correct and CTS gets corrupted
| Exactly 1 RTS gets transmitted)

TABLE 1
Durations and probabilities for the seven channel states, as functions of τ, δ, λ_g and λ_b.

No	Scenario	Duration(d_i)	Probability(p_i)
1	Idle	σ	$(1-\tau)^n$
2	RTS collision	$RTS+\delta+EIFS$	$1-(1-\tau)^n - n\tau(1-\tau)^{n-1}$
3	RTS corruption	$RTS+\delta+EIFS$	$n\tau(1-\tau)^{n-1}(1-\frac{\lambda_b}{\lambda_g+\lambda_b}e^{-\lambda_g(RTS+\delta)})$
4	CTS corruption	$RTS+\delta+SIFS+CTS+\delta+EIFS$	F(τ ,n)
5	Data corruption	$RTS+\delta+SIFS+CTS+\delta+SIFS+DATA+\delta+SIFS+ACK+\delta+DIFS$	G(τ ,n)
6	ACK corruption	$RTS+\delta+SIFS+CTS+\delta+SIFSDATA+\delta+SIFS+ACK+\delta+DIFS$	H(τ,n)
7	Successful	$RTS+\delta+SIFS+CTS+\delta+SIFS+DATA+\delta+SIFS+ACK+\delta+DIFS$	(1-(p_1+p_2+p_3+p_4+p_5+p_6))

$= n\tau(1-\tau)^{n-1}\frac{\lambda_b}{\lambda_g+\lambda_b}e^{-\lambda_g(RTS+\delta)}$ (Pr(Channel state is bad at start of CTS | Channel state was good at end of RTS)+
Pr(Channel state is good at start of CTS but becomes bad during CTS duration | Channel state was good at end of RTS)).

Now, we examine the second term above.

Pr(Channel state good at start of CTS but becomes bad during CTS duration | Channel state good at end of RTS)

= Pr(Channel state good at CTS start | It was good at end of RTS)·
Pr(Good state duration < CTS)

= Pr(Channel state good at CTS start | It was good at end of RTS) ·
$(1-e^{-\lambda_g(CTS+\delta)})$.

If $S(t)$ represents the channel state at time t, then

Pr(Channel state good at CTS start | It was good at RTS)

$= \Pr(S(t+SIFS)\text{=Good} \mid S(t)\text{=Good})$

$= B^T e^{A(SIFS)} B,$

where

$$A = \begin{vmatrix} -\lambda_g & \lambda_g \\ \lambda_b & -\lambda_b \end{vmatrix},$$

and

$$B = \begin{pmatrix} 1 \\ 0 \end{pmatrix}.$$

Therefore, we get

$$\begin{aligned} \text{F}(\tau,n) &= n\tau(1-\tau)^{n-1}\tfrac{\lambda_b}{\lambda_g+\lambda_b}e^{-\lambda_g(RTS+\delta)} \\ &\cdot((1-\Pr(S(t\text{+SIFS})\text{=Good} \mid S(t)\text{=Good})) \\ &+ \Pr(S(t+SIFS)\text{=Good} \mid S(t)\text{=Good})\cdot(1-e^{-\lambda_g(CTS+\delta)})). \end{aligned}$$

So,

$$\begin{aligned} p_4 = F(\tau,n) = n\tau(1-\tau)^{n-1}\frac{\lambda_b}{\lambda_g+\lambda_b}e^{-\lambda_g(RTS+\delta)}((1-B^Te^{A(SIFS)}B) \\ +B^Te^{A(SIFS)}B(1-e^{-\lambda_g(CTS+\delta)})). \end{aligned}$$

We similarly get expressions for G and H:

$$\begin{aligned} p_5 = G(\tau,n) = n\tau(1-\tau)^{n-1}\frac{\lambda_b}{\lambda_g+\lambda_b}e^{-\lambda_g(RTS+\delta)})(1-((1-B^Te^{A(SIFS)}B) \\ +B^Te^{A(SIFS)}B(1-e^{-\lambda_g(CTS+\delta)}))) \\ \cdot((1-B^Te^{A(SIFS)}B)+B^Te^{A(SIFS)}B(1-e^{-\lambda_g(DATA+\delta)})), \end{aligned}$$

$$\begin{aligned} p_6 = H(\tau,n) = n\tau(1-\tau)^{n-1}\frac{\lambda_b}{\lambda_g+\lambda_b}e^{-\lambda_g(RTS+\delta)})(1-((1-B^Te^{A(SIFS)}B) \\ +B^Te^{A(SIFS)}B(1-e^{-\lambda_g(CTS+\delta)}))) \\ \cdot(1-((1-B^Te^{A(SIFS)}B)+B^Te^{A(SIFS)}B(1-e^{-\lambda_g(DATA+\delta)}))) \\ \cdot((1-B^Te^{A(SIFS)}B)+B^Te^{A(SIFS)}B(1-e^{-\lambda_g(ACK+\delta)})). \end{aligned}$$

Having computed the probabilities p_i and the durations d_i for each of the seven scenarios above, we can compute the goodput as follows:

Goodput $= p_7 \cdot DATA/\Sigma_{i=1}^{7} p_i \cdot di.$

Note that Goodput has been determined as a function of τ, which still remains to be determined. We also compute the probability p that the RTS transmission is unsuccessful given that the node transmits a RTS. This probability p can be computed in terms of τ in exactly the same manner as we constructed Table 1:

$$p = q_2 + q_3 + q_4 + q_5 + q_6 \tag{1}$$

where

q_2 = probability that there is a RTS collision given that a station transmits a RTS,
q_3 = probability that there is RTS frame corruption given that a station transmits a RTS,
q_4 = probability that there is CTS frame corruption given that a station transmits a RTS,
q_5 = probability that there is DATA frame corruption given that a station transmits a RTS,
q_6 = probability that there is ACK frame corruption given that a station transmits a RTS.

Local Analysis: Computation of τ

We now compute the probability τ that a node sends an RTS in a slot as a function of p computed in the first phase. To do this, we need to model the backoff algorithm. The state of a node consists of its backoff counter value and the backoff counter stage. We represent it by an ordered pair (a,b), where a is the backoff stage in which the node is present, and b is the backoff counter value. The Markov Chain with all the states is shown in Figure 7, and is the same as the one obtained by Bianchi [1], except for the last row where we have taken the Short Retry Count into account. The last row corresponding to the Short Retry Count has a probability one transition back to the first row unlike other rows which have probability (1-p) transitions.

There are $(m + 1)$ rows where m is the station Short Retry Count (typically 7). Each row corresponds to a backoff stage. The maximum counter value for each backoff stage is doubled until we reach a stage m' (typically 5 for FHSS and 6 for DSSS). Within each row we have transitions of probability 1 from $(i, j + 1)$ to (i, j), except for the first column. The idea is that the backoff counter value for a node decreases within each backoff stage until it reaches 0. When it reaches 0 the node transmits an RTS. If the RTS transmission results in a successful transmission then the backoff stage is set to 0, and the node again chooses a counter value uniformly between 0 and CW_0. However, if the RTS transmission is unsuccessful, then the backoff stage is increased by 1, and the node chooses a backoff counter value uniformly between 0 and the maximum window size for that next row.

From the transition probabilities of the Markov Chain we now compute the steady

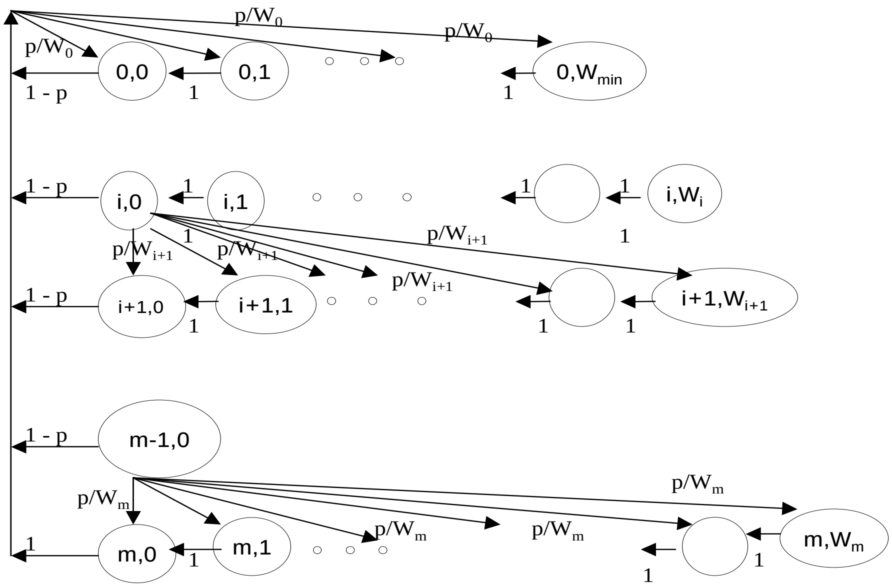

FIG. 7. *Markov Chain showing the states of a node. Each row corresponds to a backoff stage, while the columns correspond to different backoff counter values.*

state probabilities for each of the states. We assume that the system is saturated in that every node always has a packet to send. Let $\pi_{i,j}$ represent the steady state probability of the discrete time Markov Chain for the state corresponding to backoff stage i and backoff counter j.

Consider first the set of states $\{(i,0),(i,1),(i,2),...(W_i-1)\}$. This set can only be entered from state $(i-1,0)$ when there is an RTS failure which happens with probability $\pi_{i-1,0}\cdot p$. However, it can only be left from state $\pi_{i,0}$. Thus, balancing these across the cut set $\{(i,0),(i,1),(i,2),.....(i,W_i-1)\}$, we obtain

$$\pi_{i-1,0}\cdot p=\pi_{i,0}, \quad \text{for} \quad 0<i\leq m. \tag{2}$$

Hence,

$$\pi_{i,0}=p^i\pi_{0,0}. \tag{3}$$

Consider now the set of states $\{(0,k),(0,k+1),...,(0,W_0-1)\ \}$. The only way to leave the set is through the transition from $(0,k)$ to $(0,k-1)$, which happens with probability $\pi_{0,k}$. On the other hand, the only way to enter it is after a successful transmission from one of the states $\{(0,0),(1,0),(2,0),...,(m-1,0)\}$ which happens with probability $((1-p)\Sigma_{j=0}^{m-1}\pi_{j,0}+\pi_{m,0})$, followed by choosing a counter value in the range $\{k,....,W_0-1\}$ which happens with probability $\frac{W_0-k}{W_0}$. Hence,

$$\pi_{i,k}=\frac{W_i-k}{W_i}((1-p)\Sigma_{j=0}^{m-1}\pi_{j,0}+\pi_{m,0}), \quad \text{for} \quad i=0.$$

Now consider the set of states $\{(i,k),(i,k+1),...,(i,W_i-1)\}$. The only way to leave this set is through the transition from (i,k) to $(i,k-1)$ which has the flux $\pi_{i,k}$. It can be entered from the state $(i-1,0)$ after an unsuccessful RTS transmission which occurs with probability p, if the random backoff counter is chosen in the range $\{k,k+1,...,W_i-1\}$ which happens with probability $\frac{W_i-k}{W_i}$. Hence,

$$\pi_{i,k} = \frac{W_i-k}{W_i} p\pi_{i-1,0}, \quad \text{for} \quad 0 < i \le m.$$

The above two equations can be simplified using (3) to get

$$\pi_{i,k} = \frac{W_i-k}{W_i} \pi_{i,0}, \quad \text{for} \quad 0 \le i \le m,$$

and

$$\pi_{i,k} = \frac{W_i-k}{W_i} \pi_{0,0}, \quad \text{for} \quad i = m.$$

It follows that

$$\Sigma_{k=0}^{m-1} \pi_{i,k} = \frac{W_i+1}{2} \pi_{i,0} \quad \text{for} \quad 0 \le i \le m$$

and

$$\Sigma_{k=0}^{m-1} \pi_{i,k} = \frac{W_i+1}{2} \pi_{0,0} \quad \text{for} \quad i = m.$$

Using the normalization condition that the sum of all the steady state probabilities should be 1, we get

$$\pi_{0,0} = \frac{2(1-2p)(1-p)}{W(1-(2p)^{m+1})(1-p)+(1-2p)(1-p^{m+1})}, \quad \text{for} \quad m \le m',$$

$$\pi_{0,0} = \frac{2(1-2p)(1-p)}{W(1-(2p)^{m+1})(1-p)+(1-2p)(1-p^{m+1})+W2^{m'}p^{m'+1}(1-2p)(1-p^{m-m'})} \quad \text{for} \quad m > m'.$$

Note that we have computed all the $\pi_{i,j}$'s as a function of p, which is the probability that an RTS is unsuccessful given that it was attempted.

Fixed point iteration

Note now that

$$\tau = \Sigma \pi_{i,0}. \tag{4}$$

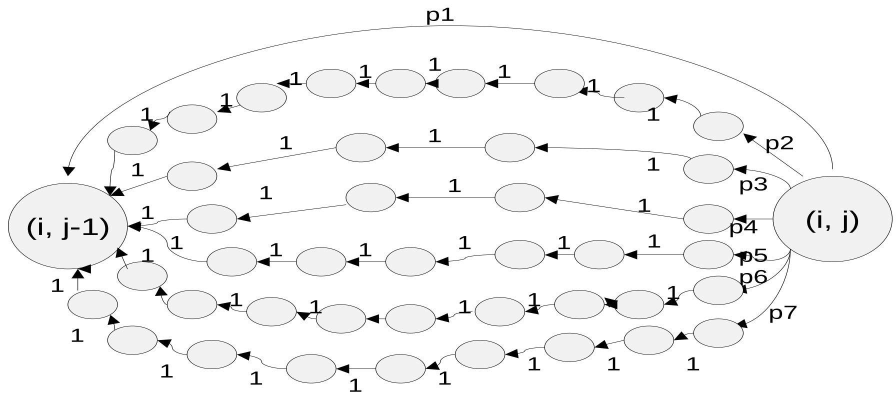

FIG. 8. *Intra row transitions in the enhanced Markov Chain used for delay analysis. Within each row of the Markov Chain shown in Figure 7, when a node's backoff counter value decreases by 1, the decrease can occur following one of the seven different channel scenarios (which have different durations in time slots) of Table 1. The Markov Chain in Figure 7, therefore, can be enhanced by adding extra states as shown above.*

However, the $\pi_{i,0}$'s themselves are functions of τ, since p is so. Thus we have computed p as a function of τ in equation (1) and computed τ as a function of p in equation (4). In order to determine the value of τ we therefore need to solve for the fixed point. We do this by way of a fixed point iteration using equations (1) and (4).

Further refinements

The model presented above was simplistic in that it assumed a single time unit for each transition from (i, k) to $(i, k-1)$. It can be refined further by adding special dummy states. The intra-row transitions and the inter-row transitions which are represented by single arrows in the original Markov Chain can be expanded and modelled as shown in Figures 8 and 9. Within each row, when a node's counter value decreases by 1 with probability 1, the decrease can occur by following one of the seven possible paths. The probabilities of each of these paths are as evaluated earlier. Similarly, when a inter-row transition occurs, there are six possibilities for the channel as depicted in Figure 9. This enhanced Markov Chain is useful for conducting a delay analysis of the system, and it also provides a fine-grained analysis of the system. Although the numerical results obtained don't differ too much from the original Markov Chain, it provides a better understanding of the system.

The numerical results were obtained by performing iterations in MATLAB. The results were compared against ns-2 simulation results. Figure 10 shows the variation of Goodput per node for five nodes, in a single hop 802.11 wireless LAN with Packet Size. The two curves were obtained using ns-2 simulations and numerical computations, and are very close to each other validating the numerical approach. Figure 11 shows three

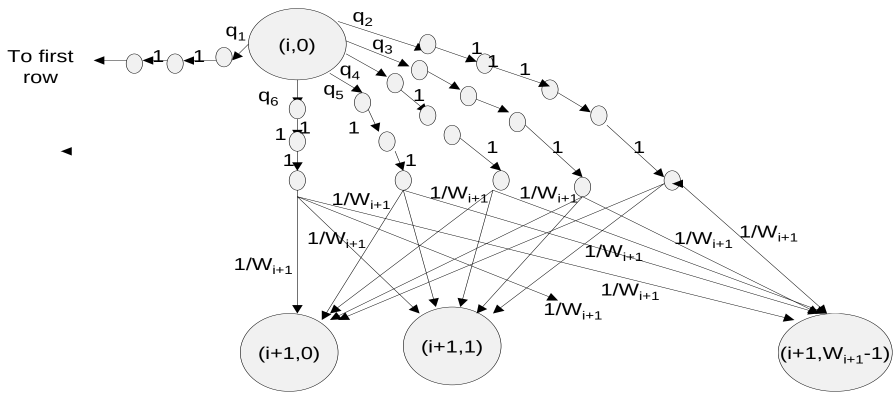

FIG. 9. *Inter row transitions in the enhanced Markov Chain used for delay analysis. When a node transmits an RTS, there are six possible channel events that can occur excluding the idle channel case addressed in Table 1. Five of them lead to unsuccessful transmission in which case the node increments its backoff counter stage represented by the next row in the Markov Chain. The durations of each of these events are different and therefore shown by adding more states corresponding to each slot in the Markov Chain.*

curves which represent the Goodput per node as the number of nodes grows in a single hop 802.11 wireless LAN, for different mean Good state and Bad state durations.

3.1.2. Delay Analysis. This section studies the mean delay by computing the time interval between two successful transmissions at a node. The enhanced Markov Chain outlined at end of the previous section provides an approach to analyzing the delay in the system. The time between two successful transmissions by a station is composed of the following: Each station spends some time in each row of the Markov Chain to get from a specific non-zero backoff counter value to backoff counter value 0. The time taken depends on the initial backoff counter value and the path followed to reach 0. Then, each station spends time in making some unsuccessful transmissions. Finally, there is the time spent in the successful transmission itself.

Let,
D = time interval between two successful transmissions,
R = the random number of RTS retransmissions a node makes before successful DATA frame transmission,
Y_i = time taken by the node to reach the backoff counter value 0 in its i^{th} (mod m) retransmission attempt,
T_s = time duration of a successful transmission.
Then,

$$D = \lfloor R/m \rfloor \Sigma_{i=0}^{m} Y_i + Y_0 + Y_1 + ... + Y_{R(mod m)} + T_s + R \cdot \Sigma_{i=2}^{6} d_i \cdot q_i.$$

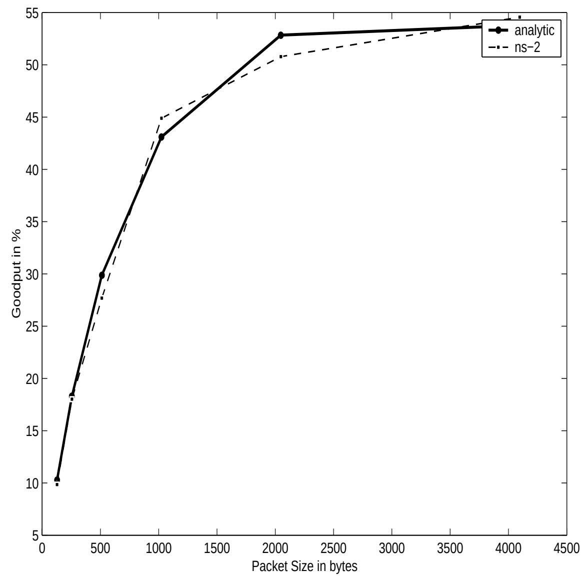

FIG. 10. *Goodput in % of capacity (per node) versus Packet Size in bytes (capacity is 11 Mbps). The Good and Bad state durations (in mean) were assumed to be 100ms and 10ms respectively. A total of 5 nodes in a collocated topology were considered. The analysis takes into account the time taken for transmitting the physical layer PLCP header which is transmitted at the basic rate of 1 Mbps, while the rest of the frame is assumed to be transferred at 11 Mbps. The SIFS and Slot times are assumed to be 28 μs and 50 μs, respectively.*

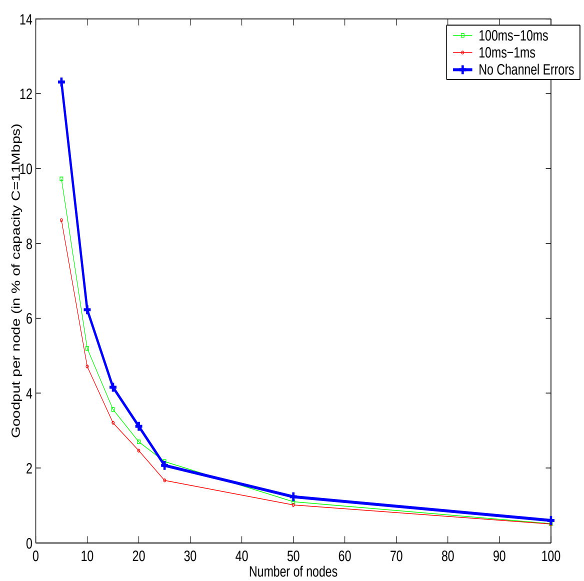

FIG. 11. *Goodput in % of capacity per node versus Number of nodes (capacity is 11 Mbps). The three curves represent scenarios corresponding to mean Good and Bad state durations of 100ms-10ms, and 10ms-1ms, and without error scenarios, respectively. Using a best-fit straight line to the plot of the logarithm of the Goodput versus the number of collocated nodes n, it was observed that Goodput is proportional to $n^{-1.08}$*

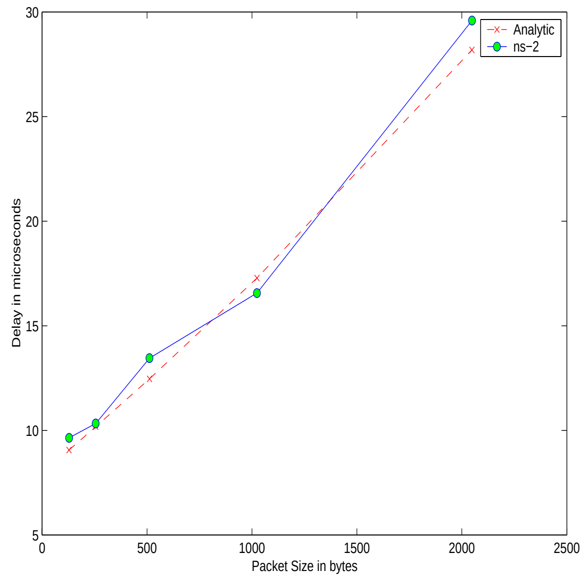

FIG. 12. *Delay in microseconds versus Packet Size in bytes using ns-2 and numerical analysis.*

Since the value of the backoff counter is chosen uniformly as a random integer between 0 and $W_i - 1$, and the probabilities and durations of each of the seven scenarios are given by p_i and d_i respectively, we have

$$E[Y_i] = \frac{W_i - 1}{2} \Sigma_{k=1}^{7} d_k p_k.$$

The number of retransmissions depends upon q_1, which is the probability that a transmission is successful given that a station transmits an RTS as shown in Figure 9. The number of retransmission attempts R is geometrically distributed:

$$Pr(R = i) = q_1 (1 - q_1)^i, \quad for \quad 0 \leq i \leq m.$$

Combining the above equations we get,

$$E[D] = \Sigma_{i=0}^{m} P(R = i) E(D|R = i).$$

Figure 12 shows the variation of delay against different packet sizes using ns-2 simulation and the analysis presented above.

3.2. Performance analysis of Multi-hop 802.11 wireless LAN. Now we turn to the problem of performance evaluation in multi-hop scenarios. Analysis of 802.11 wireless LANs in the multi-hop case differs significantly from the single hop case due to two main reasons:

1. First, each node hears different events on the channel, unlike the collocated case where there is a common view of the wireless channel. So a common analysis that applies to all nodes similar to the collocated case first phase analysis where Goodput is computed as a function of τ, is not feasible.
2. Second, with a general channel model, the possible channel events in the multi-hop case are more than just the seven cases for the collocated case. We need to consider several scenarios as there can be DATA-RTS collisions, as well as RTS-RTS collisions as shown in Figure 13. Also, it is possible to have partial overlap of frames as the nodes cannot listen to all nodes.

3.2.1. Approach. We use the same model assumptions for the multi-hop case except that:

1. The wireless channel is assumed to be perfect and there are no bit errors due the wireless channel.
2. The back-off algorithm does not use a back-off stage or exponential doubling of the maximum window size.

The assumptions made above are solely for ease of analysis. They can be incorporated as earlier done for the single hop case. Under these assumptions, there are three possible events that can occur when a node transmits an RTS: The RTS frame transmission is unsuccessful due to a collision with some other frame, the RTS frame is successful but the DATA frame transmission is unsuccessful due to a collision with some frame, or the four phase handshake is successful and the DATA frame is successfully acknowledged.

In this model we are not considering frame corruptions due to wireless bit errors. The DCF protocol ensures that CTS frame transmission will be successful if the RTS frame is successfully delivered to the destination and there are no bit errors. This is because successful RTS frame transmission silences all the nodes in the neighborhood of the source either for a duration specified in the duration field of the RTS or for EIFS time (if a collision occurs) which is large enough to transmit a CTS. Therefore, the CTS frame cannot collide with any frame at the source. Using a similar argument, it can be concluded that a successful DATA frame transmission ensures a successful ACK frame transmission. Thus, we cannot have collisions involving CTS and ACK frames. Also, DATA-DATA frame collisions are not feasible.

However, there can be DATA-RTS frame collisions even after a successful RTS-CTS frame exchange. The DATA-RTS frame collisions occur due to the presence of hidden terminals in the multi-hop scenario. Figure 12 depicts the two possible sequences of events that result in DATA-RTS collisions.

The analysis of goodput in the multi-hop case is a local analysis unlike the global analysis for the single hop case. We construct a Markov Chain to model the different states of a node. The possible states of a node are its back-off counter stages when

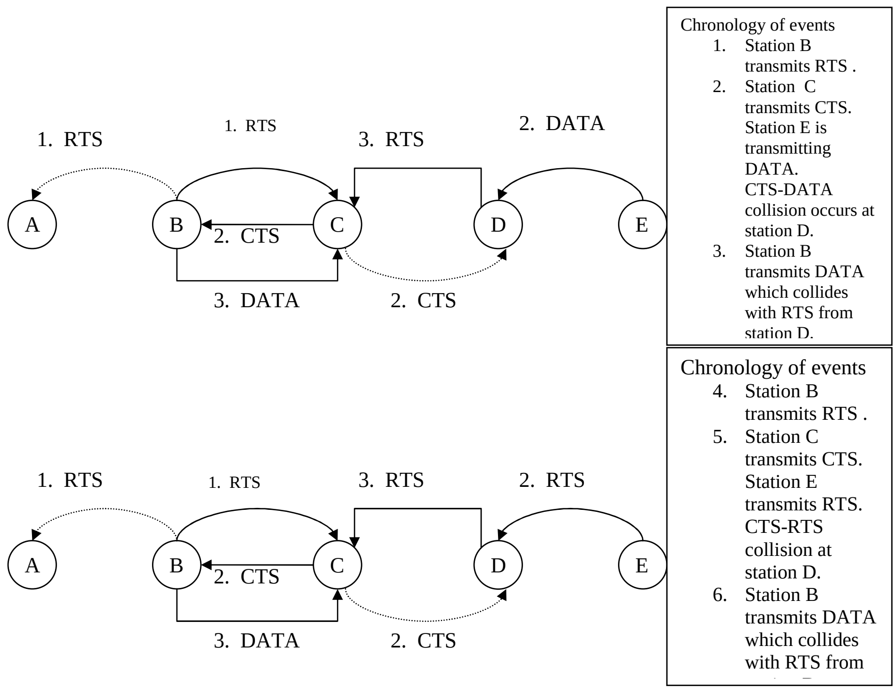

FIG. 13. *Two scenarios that result in DATA-RTS frame collisions.*

it is just listening to the channel, and the states corresponding to each of the three possible scenarios after an RTS transmission. Figure 14 shows the Markov Chain structure. There are l_W or $(W_{max}+1)$ backoff states, l_R RTS unsuccessful states, l_U Unsuccessful transmission states, and l_S successful states. The backoff state i, corresponds to the node being in the waiting state with backoff counter value i. The RTS unsuccessful states correspond to an unsuccessful RTS transmission attempt by the node. The unsuccessful transmission states correspond to unsuccessful DATA frame transmission after a successful RTS transmission due to a DATA-RTS frame collision. The successful transmission states correspond to a successful four phase handshake. The probabilities P_{WR}, P_{WU}, and P_{WS} represent the transition probabilities of the three possible scenarios after an RTS frame transmission. p is the probability of a node retaining the same backoff counter value after a slot.

The analytical relations for the goodput analysis are of two types, Local relations and Topology dependent relations. Local relations are determined by the structure of the Markov Chain above. Topology dependent relations vary for each topology. The interaction between nodes in the multi-hop case depends on the topology, e.g., the probability that an RTS frame transmission will result in a collision depends on the number of neighboring nodes, and the probability that a neighboring node will transmit a frame during the RTS frame transmission. In the next section, we present such relations for a ring topology.

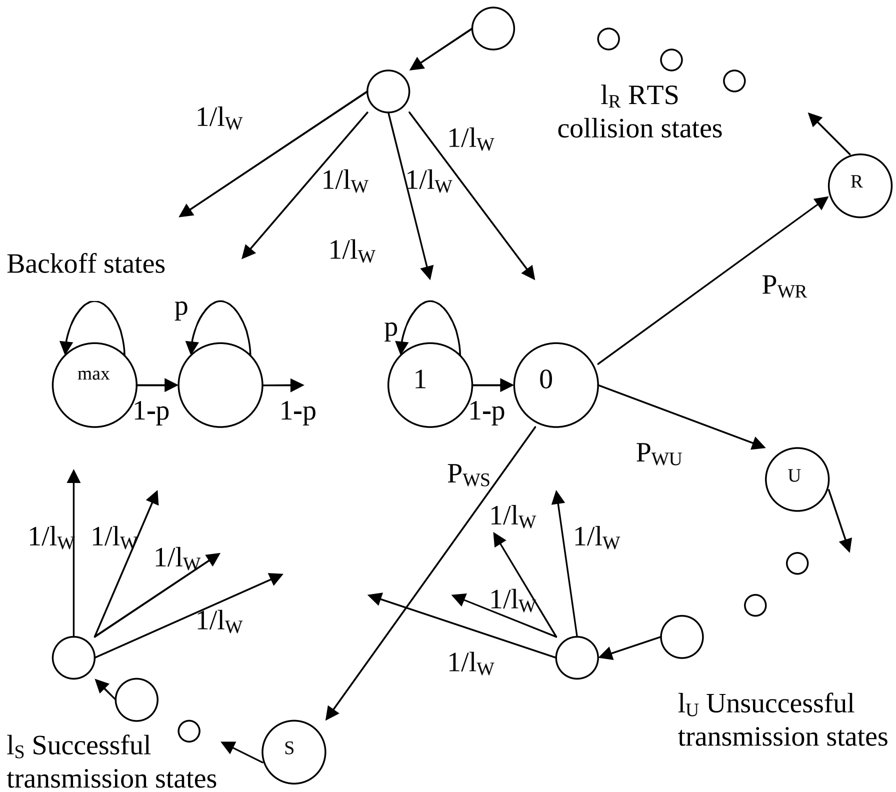

FIG. 14. *Markov Chain for a node in the multi-hop scenario. A node can be either in one of the backoff states, RTS Unsuccessful states, Successful transmission states, or Unsuccessful after RTS success states. There are self-loop transitions with probability p in the backoff states which model the varying number of slots a node spends with a particular backoff counter value. After transmitting a RTS which happens when the backoff counter reaches 0, a node might have a successful four phase handshake, an unsuccessful RTS transmission, or an unsuccessful DATA frame transmission after a successful RTS transmission. The transition probabilities for the three possible events are represented by* P_{WR}, P_{WU}, *and* P_{WS}.

Using balance across cut sets in the Markov Chain shown in Figure 13, we derive the local relations. The transition probabilities P_{WR}, P_{WS}, P_{WU} and p together determine the steady state probabilities for each of the states in the Markov Chain.

Let,
Π_i = the steady state probability of the backoff state with counter value i (recall there is only one backoff stage),
Π_R = the steady state probability of being in any of the RTS collision states,
Π_U = the steady state probability of being in any of the unsuccessful transmission states,
Π_S = the steady state probability of being in any of the successful transmission states.
Then,

$$\Pi_R = \Pi_0 P_{WR},$$

$$\Pi_S = \Pi_0 P_{WS},$$

$$\Pi_U = \Pi_0 P_{WU},$$

Also,

$$\Pi_{max} = \frac{\Pi_0}{(1-p)(W_{max}+1)},$$

$$\Pi_i = \frac{\Pi_0(W_{max}-i+1)}{(1-p)(W_{max}+1)}.$$

The sum of steady state probabilities of each of the states should be 1. Therefore from the above relations, we get

$$\Pi_0 = (1 + l_R P_{WR} + l_U P_{WU} + l_S P_{WS} + \frac{W_{max}}{(1-p)})^{-1}.$$

Topology dependent relations determine the transition probabilities P_{WR}, P_{WS}, P_{WU} and p in terms of the steady state probabilities of the Markov Chain. These relations use a number of simplifying approximations and depend upon the specific topology. We then use fixed point iteration to determine the stationary steady state probabilities, and then use them to determine the throughput.

3.2.2. Goodput Analysis for Ring Topology. We consider a ring of nodes where each node is transmitting to its neighbor (say in the clockwise direction), as shown in Figure 15. Station B's RTS transmission collides with some other frame if hidden station D is transmitting a frame in the slot when RTS transmission begins, or there is a collision after k slots where $k < l_R$. The probability that station D is transmitting some frame in a particular slot is approximated by the expression $(l_R\Pi_R + l_U\Pi_U + l_S\Pi_S)$, as it is the fraction of time a station is transmitting. The probability that a collision occurs after one slot is given by Π_0, after two slots is given by $\Pi_1.(1-p)$ and after three slots is given by $(\Pi_2(1-p)^2 + \Pi_1 p(1-p))$, and so on. By substituting the values for Π_i derived above in each of these terms, we have observed that they can be well approximated by Π_0. Adding all such terms, we get

$$P_{WR} = (l_R\Pi_R + l_U\Pi_U + l_S\Pi_S + \Pi_0(l_R - 1)). \tag{5}$$

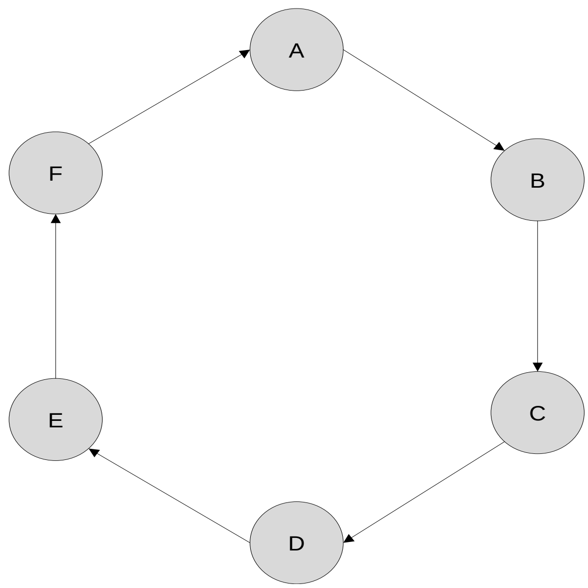

FIG. 15. *The Ring topology. Each node can listen to two other nodes. Each node is sending traffic to its neighbor in the clockwise direction as shown by the arrows.*

Computation of P_{WU} requires considering the two cases depicted in Figure 13. There will be a DATA-RTS collision only if the backoff time for station D is smaller than the slots left during the DATA transmission.

We first compute the probability that the total backoff time for node D is less than k slots:

$$Pr(BackoffTime \leq k) = \frac{1}{l_W(2 - p^{k-1})}.$$

Also, we compute the number of slots left, s, as

$$s = \frac{(DATA - EIFS + SIFS)}{\sigma}.$$

Therefore, the probability Pr(CTS-RTS) for a DATA-RTS collision after the CTS frame collides with a RTS frame at node D (the second case depicted in Figure 10), is computed by adding the probabilities for all possible cases of DATA-RTS collisions. $l_R \Pi_R$ is the probability that node E is in one of the possible RTS transmission stages when CTS transmission begins. $(2 - p^{s-1})/l_W$ is the probability that the backoff time for node D is less than s slots. Therefore, we get the first term in the equation below as the product of these two. The second term is obtained by adding the terms for cases when RTS transmission by node E starts in the first, second, and till the last slot of CTS transmission, and then multiplying those with the corresponding terms for the backoff time being less than or equal to $(s-1)$, $(s-2)$, and so on,

$$Pr(CTS - RTS) = \frac{l_R \Pi_R (2 - p^{s-1})}{l_W} + \frac{\Pi_0 (2(s-2) - (1 - p^{s-2})/(1-p))}{l_W}.$$

The probability Pr(CTS-DATA) for a DATA-RTS collision after the CTS frame collides with a DATA frame (the first case depicted in Figure 10) is given by the following expression; the computation is done in a similar manner as for the CTS-RTS case:

$$Pr(CTS - DATA) = \frac{l_C(\Pi_S + \Pi_U)(2 - p^{s-1})}{l_W} + \frac{(\Pi_S + \Pi_U)(2(s-2) - (1 - p^{s-2})/(1-p))}{l_W}.$$

Using the above expressions, we can write topology dependent equations for the transition probabilities P_{WU} and P_{WS} as shown below. The transmission will be unsuccessful even after a successful RTS transmission if DATA-RTS collisions occur. So,

$$P_{WU} = (1 - P_{WR})(Pr(CTS - DATA) + Pr(CTS - RTS)), \tag{6}$$

$$P_{WS} = (1 - P_{WR})(1 - (Pr(CTS - DATA) + Pr(CTS - RTS))). \tag{7}$$

The loop probability p in the Markov Chain depends upon the fact that each station has two neighbors. The probability p depends upon the probability q that a slot is observed to be idle by a station. The probability q that a slot is observed to be idle is given by the probability that both neighbors do not transmit anything during the slot,

$$q = (1 - (l_R \Pi_R + l_U * \Pi_U + l_S * \Pi_S))^2.$$

Also, $1/(1-p)$ is the average or expected duration between a backoff counter decrement and is given as follows:

$$\frac{1}{1-p} = q + (1-q)(l_R P_{WR} + l_U P_{WU} + l_S P_{WS}). \tag{8}$$

Now, we consider the equations (5),(6),(7) and (8). We perform a fixed point iteration assuming some initial values for p, P_{WR}, P_{WU} and P_{WS}, and the local relation for Π_0, to determine their values.

Finally, we obtain Goodput as

$$Goodput = DATA * \Pi_S / \sigma.$$

A similar analysis is done for the mesh topology by using topology dependent relations specific to the mesh topology.

Figures 16 and 17 compares the analytical results with the ns-2 simulation results for the ring and mesh topology, respectively. We use CW values of 31 and 63 in the analytical model and obtain a lower and upper bound on the ns-2 goodput. Note that the multi-hop analysis does not take into account doubling of contention window CW. Therefore, using a contention value of 31 will result in a more optimistic estimate of throughput. Similarly, using CW value of 63 results in a pessimistic estimate of throughput.

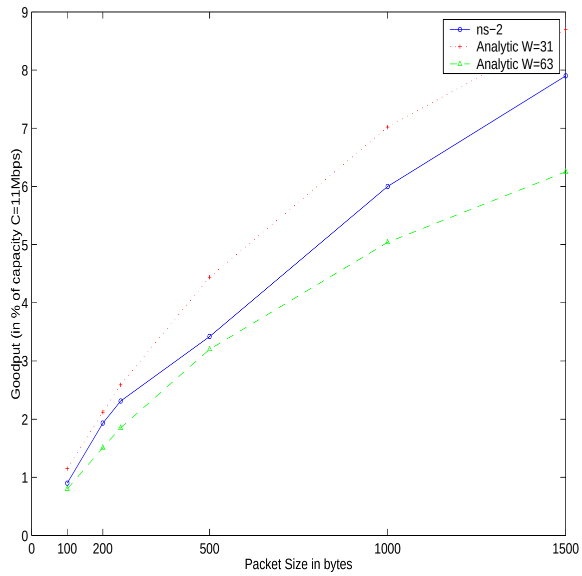

FIG. 16. *Goodput in % of capacity Versus Packet Size for a Ring topology with 6 nodes (capacity is 11 Mbps). The middle curve is obtained using ns-2 simulations. The top and bottom curves represent numerical results with CW values of 31 and 63 respectively, and bracket the true values.*

4. Concluding Remarks. We have described the IEEE 802.11 medium access control mechanism and presented a goodput analysis and a delay analysis for a single hop 802.11 wireless LAN. The goodput analysis extends previous works on analysis of 802.11 wireless LANs, specifically [1], as it takes into account wireless channel errors and short retry count defined by the protocol. The numerical results obtained matched pretty well with ns-2 simulation results. The delay analysis uses an enhanced Markov Chain which helps in better understanding of the system.

Analyzing the 802.11 MAC protocol in multi-hop scenarios is more involved and no previous analyses exist, to our knowledge. We have presented a methodology for conducting a goodput analysis of 802.11 MAC in multi-hop networks. The approach requires forming local relations which are topology independent, as well as topology dependent relations. We have presented the application of the approach to a ring topology and constructed the topology dependent relations for the ring topology. A similar approach was also used for analyzing a mesh topology. The numerical results were validated against ns-2 simulation results.

There are lots of avenues for future work. The multi-hop methodology needs to be applied to more general heterogeneous topologies. The approach requires constructing topology specific relations. Alternative approaches for constructing such topology dependent relations are worth investigating.

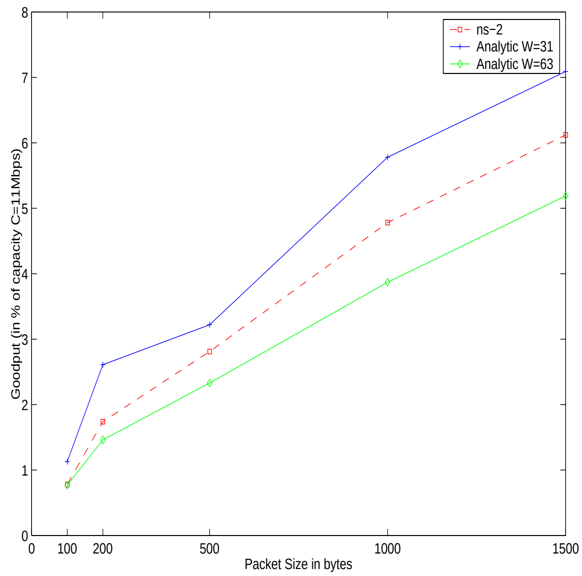

FIG. 17. *Goodput in % of capacity Versus Packet Size for a Mesh topology (capacity is 11Mbps). The middle curve is obtained using ns-2 simulations. The top and bottom curves represent numerical results with CW values of 31 and 63 respectively, and bracket the true values.*

REFERENCES

[1] G. BIANCHI, *Performance Analysis of the IEEE 802.11 Distributed Co-ordination Function*, IEEE Journal on Selected Area in Communications V18, N3, 2000.

[2] F. CALI, M. CONTI, AND E. GREGORI, *IEEE 802.11 wireless LAN: Capacity analysis and protocol enhancement*, INFOCOM 98 San Francisco, CA, Mar 1998.

[3] *IEEE Standard for Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*, Nov. 1997. P802.11.

[4] H. S. CHHAYA AND S. GUPTA, *Performance modeling of asynchronous data transfer methods of IEEE 802.11 MAC protocol*, Wireless Networks, 3(1997)), pp. 217–234.

[5] T. S. HO AND K. C. CHEN, *Performance evaluation and enhancement of the CSMA/CA MAC protocol for 802.11 wireless LANs*, in: Proc. IEEE PIMRC, Taipei, Taiwan, Oct. 1996, pp.392–396.

[6] K. HUANG, K. CHEN, *Interference Analysis of Nonpersistent CSMA with Hidden Terminals in Multicell Wireless Data Networks*, Proc. PIMRC 1995, Toronto, pp. 907–911, Sept. 1995.

[7] BLUETOOTH SIG, Specification of the Bluetooth system, Core Version 1.1. 1 February 22, 2001, available at http://www.bluetooth.com/.

[8] ETSI, Radio Equipment and Systems (RES); HIgh Performance Radio Local Area Network (HIPERLAN), Type 1 Functional Specification, ETS pp.300-652 (Ed.1), August 1996.

[9] The HomeRF Working Group, http://www.homerf.org.