New Insights From a Fixed-Point Analysis of Single Cell IEEE 802.11 WLANs

Anurag Kumar, *Fellow, IEEE*, Eitan Altman, Daniele Miorandi, and Munish Goyal

***Abstract*—We study a fixed-point formalization of the well-known analysis of Bianchi. We provide a significant simplification and generalization of the analysis. In this more general framework, the fixed-point solution and performance measures resulting from it are studied. Uniqueness of the fixed point is established. Simple and general throughput formulas are provided. It is shown that the throughput of any flow will be bounded by the one with the smallest transmission rate. The aggregate throughput is bounded by the reciprocal of the harmonic mean of the transmission rates. In an asymptotic regime with a large number of nodes, explicit formulas for the collision probability, the aggregate attempt rate, and the aggregate throughput are provided. The results from the analysis are compared with *ns2* simulations and also with an exact Markov model of the backoff process. It is shown how the saturated network analysis can be used to obtain TCP transfer throughputs in some cases.**

***Index Terms*—CSMA/CA, performance of MAC protocols, wireless networks.**

I. Introduction

We are concerned in this paper with the situation in which there are several IEEE 802.11 compliant nodes within such a distance of each other that only one transmission can be sustained at any point of time. We call these *single-cell* networks. Our discussion covers *ad hoc networks* and also *infrastructure networks*, in which an AP acts as a conduit between the wireless network and a wired "infrastructure." Our analysis is limited to the situation in which all nodes use the RTS/CTS based distributed coordination function (DCF) without the QoS extensions (as in IEEE 802.11e) (but see [9] for our extensions of the work in the present paper).

Each node may have several physical *connections or associations* with several other nodes. On each such connection the sustainable *physical* transmission rate may be different. Between each such pair of nodes there are *flows* whose throughput performance we are concerned with. It is assumed throughout this paper that all flows are infinitely back-logged at their transmitters; i.e., there are always packets to transmit when a node gets a chance to do so.

In such a scenario, we are interested in obtaining quantitative formulas and qualitative insights via a stochastic analysis of the way that the IEEE 802.11 CSMA/CA protocol allocates the wireless medium to the node transmitters. Our approach is to begin with a key approximation made by Bianchi [3]. This leads to a fixed-point equation, which can be expected to characterize the operating points of the system. This fixed-point equation is our point of departure. We simplify and generalize the analysis leading to the fixed-point equation. We then establish a simple, and practically appealing, condition for the uniqueness of the fixed point in this more general framework. Some simple observations lead to throughput formulas for the overall network and for the individual flows. These formulas allow us to recover the well known observation that the slowest transmission rate dominates the throughput performance. We also analyze the fixed point in the asymptotic regime of a large number of nodes and find explicit formulas for the collision probability, the channel access rate and the network throughput. A key parameter in the protocol is the backoff multiplier, whose default value in the IEEE 802.11 MAC standard is 2; our asymptotic analysis provides some insights into the role of the backoff multiplier.

We provide *ns2* simulation results for the collision probabilities and compare these with results obtained from the fixed-point analysis. We also provide results from an exact Markov chain model for the backoff process and also compare these results with those from the fixed-point analysis.

As already pointed out, the above described modeling assumes that there are always packets backlogged on every connection. Such a *saturation assumption* is a common simplification and is useful in the following ways. In some situations, it has been formally proved (see, for example, [1] and [6]) that the saturation throughput provides a sufficient condition for stability of the queues; i.e., if, at each queue, the arrival rate is less than the saturation throughput, then the queues will have a proper, joint stationary distribution. In this paper, we also apply the saturation throughput analysis to provide an analysis for TCP-controlled file transfer throughputs in certain local area network scenarios.

The most popular model for IEEE 802.11 networks, and one that has led to many applications and extensions, is the one reported in [3]. Another analysis, that also incorporates the feature of adapting the backoff parameters, has been reported in [4]. The recent paper in [2] is one of the many that have reported a throughput "anomaly" in IEEE 802.11 networks; i.e., if the network has low-speed connections, even the high-speed connections experience throughput no better than what is obtained by the low-speed connections.

Manuscript received March 11, 2005; revised December 21, 2005; approved by the IEEE/ACM Transactions on Networking Editor R. Mazumdar. This is an extended version of a paper that appeared in IEEE Infocom 2005. This work was supported by the Indo-French Centre for Promotion of Advanced Research (IFCPAR) under Research Contract 2900-IT.

A. Kumar and M. Goyal are with the Department of Electrical Communication Engineering, Indian Institute of Science, Bangalore 560 012, India (e-mail: anurag@ece.iisc.ernet.in; munish@ece.iisc.ernet.in).

E. Altman is with INRIA, BP93 06902, Sophia-Antipolis, France (e-mail: eitan.altman@sophia.inria.fr).

D. Miorandi is with CREATE-NET, Trento 38100, Italy (e-mail: daniele.miorandi@create-net.org).

Digital Object Identifier 10.1109/TNET.2007.893091

1063-6692/$25.00 © 2007 IEEE

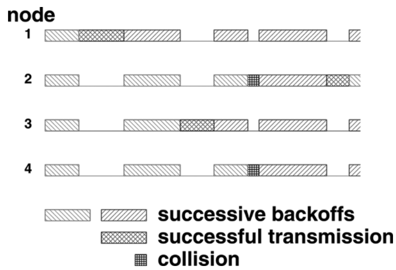

Fig. 1. Evolution of the backoff periods and channel activity for four nodes. Backoffs are interrupted by channel activity, i.e., packet transmissions and RTS collisions.

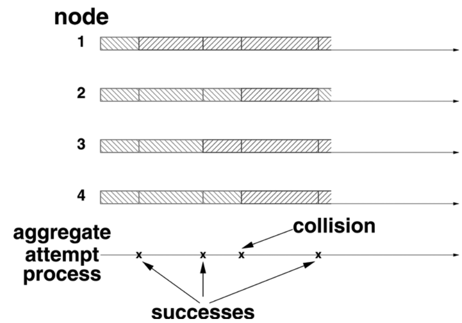

Fig. 2. After removing the channel activity from Fig. 1, only the backoffs remain. Shown at the bottom is the aggregate attempt process on the channel, with three successes and one collision.

The paper is organized as follows. In Section II, we provide the key observation and approximation on which the analysis is based. In Section III, we analyze the backoff process in a fairly general setting. The fixed-point equation is provided in Section IV and analyzed in Section V; a validation through an exact solution of a Markov model is given in Section V-B. In Section VI, the throughput formulas are provided. The asymptotic analysis is developed in Section VII. An application of the results to the analysis of TCP is given in Section VIII and the paper ends with a concluding section. Some proofs are provided inline and others are in the Appendix. Some details, not provided in this paper (including the proof of Theorem 5.2), can be found in [5] and [10].

II. Key Observation and an Approximation

A. *Sufficiency of the Analysis of the Backoff Process*

We begin by extracting from a description of the system the key modeling abstractions that will allow us to develop the analysis. Fig. 1 shows the evolution of the system for 4 nodes; shown are the backoffs, the transmissions and collisions. In the IEEE 802.11 standard, the backoff durations are in multiples of a standardized time interval called a *slot* (e.g., 20 μs in IEEE 802.11b). However, this discrete nature of the backoffs does not affect the following argument. When a node completes its backoff (for example, node 1 is the first to complete its backoff in Fig. 1), it seeks a reservation of the channel by sending an RTS packet. If no other node completes its backoff before hearing this transmission then the RTS effectively reserves the channel for the first node. There follows a CTS from the intended recipient of the RTS, and then there follows a packet transmission and a MAC level ACK. This ends the reservation period and the node that transmitted the packet samples a new backoff interval. Note that we assume throughout that nodes always have packets to transmit; i.e., *all the transmission queues are saturated*.

If the RTS collides with that of another node (note that we do not model the phenomenon of packet capture), then after fully transmitting their RTSs, each transmitting node waits for a time interval $\mathrm{SIFS}+T_{\mathrm{CTS}}+\mathrm{DIFS}$, where T_{CTS} is the time required to transmit a CTS (at the control rate of 2 Mb/s), before returning to the backoff state. For example, in Fig. 1, nodes 2 and 4 collide after the first two attempts (by nodes 1 and 3, respectively) are successful. The other nodes, not involved in the collision, and not being able to decode anything, listen to the channel activity until the end of the RTS transmissions, and then wait for an amount of time equal to $\mathrm{EIFS}\ (=\ \mathrm{SIFS}\ +\ T_{\mathrm{ACK}}\ +\ \mathrm{DIFS})$. Since an ACK and a CTS have the same number of bits, after a collision all nodes resume their backoff phases after an amount of time equal to the transmission time of an RTS plus a fixed time (equal in each case to an EIFS).

If attempts to send the packet at the head-of-the-line (HOL) meet with several successive failures, this packet is discarded. By our assumption of saturated queues, there is always another packet waiting to be sent by the upper layers: either the same packet or the next one in line.

We see from the figure that, when any node has reserved the channel or whenever there is a collision, all other nodes freeze their backoff timers. We also notice that the evolution of the channel activity after an attempt is deterministic. It is either the time taken for a transmission or for a collision. If there is a transmission, then the time depends on which node captures the channel. The latter dependence comes about because the transmission time of a packet depends on the transmission rate and, hence, on the transmitting node.

Since all nodes freeze their backoffs during channel activity, the total time spent in backoff up to any time t, is the *same* for every node. With this observation, let us now look at Fig. 2 which shows the backoffs of Fig. 1 with the channel activity removed. Thus, in this picture, "time" is just the cumulative backoff time at each node. In the IEEE 802.11 standard, the backoffs are multiples of the slot time. A success occurs if a single backoff ends at a slot boundary, and a collision occurs when two or more backoffs end at a slot boundary. The nodes could have different backoff parameters (the mean backoff intervals, how these are varied in response to collisions and successes, and the number of retries of a packet). It is clear, however, that the (random) sequence in which the nodes seek turns to access the channel and whether or not each such attempt succeeds depends only on the backoff process shown in Fig. 2. It is, therefore, sufficient to analyze the backoff process in order to understand the channel allocation process. The saturation assumption is crucial here since, with this assumption, we do not have to take care of any external packet arrivals that may occur during channel activity periods.

Thus, in summary, we can delete the channel activity periods, and we are left with a "conditional time" which we will call *backoff time*. We will analyze the backoff process conditioned on being in backoff time. It will then be shown how this analysis can be used to yield the desired performance measures over all time.

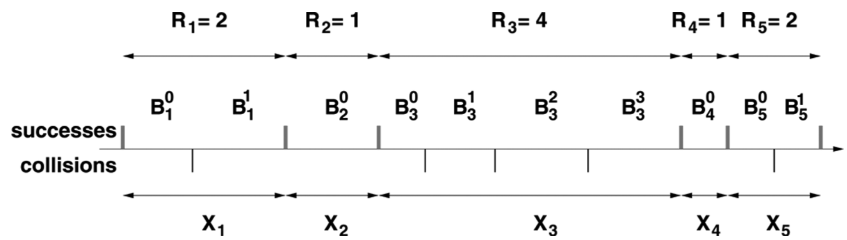

Fig. 3. Evolution of the backoffs of a node. Each attempted packet starts a new backoff "cycle."

B. Key Approximation

Throughout the rest of the paper *we assume that all the nodes use the same backoff parameters*. Hence, the backoff process shown in Fig. 2 is symmetric over the nodes. We call this the **homogeneous case** to distinguish it from the **nonhomogeneous** case in which different nodes may use different backoff parameters, as, for example, proposed in the IEEE 802.11e standard (see [8] and [9]).

In Fig. 2, we also show the aggregate sequence of successes and collisions. In general, this is a complex process, and it is also clear that the success and collision processes of the various nodes are coupled and strongly correlated. In Section V-B, we will describe an exact Markov chain model for the joint backoff process of the nodes, but this model is analytically intractable. The following key approximation is made in [3].

The Decoupling Approximation: Let β denote the long run average backoff rate (*in backoff time*) for each node. By the fact that all nodes use the same backoff parameters, and by symmetry, it is assumed that all nodes achieve the same value of β. Let there be n contending transmitters, and consider a given node. The decoupling approximation is to assume that the aggregate attempt process of the other $(n-1)$ nodes is independent of the backoff process of the given node. In IEEE 802.11, the backoff evolves over slots; hence, a discrete time model (embedded at slot boundaries) can be adopted. Then the approximation is the following. 1) The "influence" of the other nodes on a tagged node is modeled via the decoupling approximation. Attempts by a tagged node over slots experience the collision probability γ. For a given collision probability, this yields one equation $\beta = G(\gamma)$ [see (1)]. 2) The nodes are assumed to attempt in each slot with a constant (state independent) probability equal to the average attempt rate, β. Then, conditional on a tagged node attempting, the number of attempts by other nodes is binomially distributed. This yields the other ("coupling") equation $\gamma = \Gamma(\beta)$ [see (2)]. When these equations are put together, we obtain the desired fixed-point equation. It might be expected that such a decoupling approximation should work well when there is a large number of transmitters accessing the channel.

III. Analysis of the Backoff Process

We generalize the backoff behavior of the nodes and define the following backoff parameters.

$K :=$ At the $(K+1)$th attempt, either the packet succeeds or is discarded.

$b_k :=$ The mean backoff duration (in slots) at the kth attempt for a packet, $0 \le k \le K$.

Since we are limiting ourselves to the homogeneous case, these parameters are the same for all the nodes.

In Fig. 3, we show the evolution of the backoff process for a single node. There are R_j attempts until success for the jth packet (no case of a discarded packet is shown in this diagram), and the sequence of backoffs for the jth packet is $B_j^{(i)}$, $0 \le i \le R_j - 1$. Thus, the total backoff for the jth packet is given by $X_j = \sum_{i=0}^{R_j-1} B_j^{(i)}$ with $\mathsf{E}(B_j^{(i)}) = b_i$. We observe that the sequence X_j, $j \ge 1$, are renewal lifetimes. Hence, viewing the number of attempts R_j for the jth packet as a "reward" associated with the renewal cycle of length X_j, we obtain from the renewal reward theorem that the backoff rate is given by $\mathsf{E}(R)/\mathsf{E}(X)$. Now let γ be the collision probability seen by a node, i.e.,

$$\gamma := \quad \Pr\text{(an attempt by a node fails because of a collision).}$$

Since the backoff behavior of all the nodes is the same, the collision probability is the same for all the nodes. By the approximation made in Section II, the successive collision events are independent. It is then easily seen that

$$\begin{aligned} \mathsf{E}(R) &= 1 + \gamma + \gamma^2 + \cdots + \gamma^K \\ \mathsf{E}(X) &= b_0 + \gamma b_1 + \gamma^2 b_2 + \cdots + \gamma^k b_k + \cdots + \gamma^K b_K \end{aligned}$$

which yields the following formula for the attempt rate for a given collision probability γ:

$$G(\gamma) := \frac{1 + \gamma + \gamma^2 + \cdots + \gamma^K}{b_0 + \gamma b_1 + \gamma^2 b_2 + \cdots + \gamma^k b_k + \cdots + \gamma^K b_K}. \quad (1)$$

Note that, since the backoff times are in slots, the attempt rate $G(\gamma)$ is in *attempts per slot*.

Remarks 3.1:

1) Note that the *distribution* of the backoff durations does not matter. Also, observe that the above analysis remains unchanged whether the backoff distributions are discrete (i.e., the backoffs evolve over slots) or are continuous.
2) In the backoff model considered in [3], $K = \infty$; further, there is an $m \ge 1$ such that $b_k = ((2^k CW_{\min} \pm 1)/2)$ slots, for $0 \le k \le m-1$, and $b_k = ((2^m CW_{\min} \pm 1)/2)$ slots, for $k \ge m$. Here, $CW_{\min}$ is a positive integer (2^5 in the IEEE 802.11 standard). Substituting these into the expression for $G(\gamma)$ in (1) yields

$$G(\gamma) = \frac{2(1-2\gamma)}{(1-2\gamma)(CW_{\min} \pm 1) + \gamma CW_{\min}(1-(2\gamma)^m)}$$

attempts per slot, which is the same as in the paper [3]. Note that the $\pm$ alternatives arise depending on whether we take the backoff to be uniformly distributed over $[1, 2, \cdots, CW]$ or over $[0, 1, \cdots, CW-1]$. Evidently, the

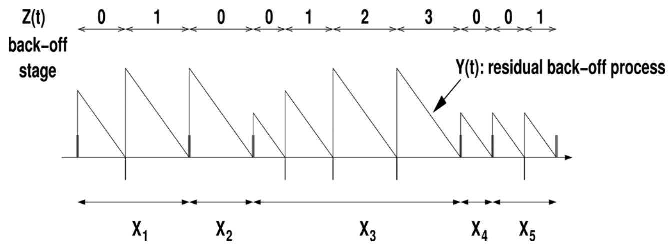

Fig. 4. Evolution of the backoff stage ($Z(t)$) and the residual backoff time ($Y(t)$) for the case in which the backoffs are continuous variables.

uniform distribution of backoff durations plays no role in the final results in [3].

3) A more detailed evolution of the backoff process in Fig. 3 is shown in Fig. 4, where, at each time t, the residual backoff duration $Y(t)$ is also shown. The process $Z(t)$ is the backoff *stage* the node is in. Thus, if $K = 7$, $Z(t) = 3$, and $Y(t) = 5$, then after five time units, the current backoff ends. If there is a collision, $Z(t)$ changes to 4, and a backoff with mean b_4 is sampled from the specified backoff distribution (uniform in the standard). If $Z(t) = 7$, then at the end of the current backoff, irrespective of whether there is a collision or a success, the next backoff has mean b_0 and is sampled from the specified distribution. It is clear that the process $(Z(t), Y(t))$ is Markov. The point is that it is not necessary to analyze this Markov chain, which is essentially what is done in [3]. Let Z_k, $k \geq 0$, denote the process $Z(t)$ embedded at the attempt instants (the instants corresponding to the vertical sides of the triangles in Fig. 4). Then Z_k is an embedded Markov chain. Further, Z_k and the successive backoff intervals (the bases of the triangles) constitute a Markov renewal process. It is well known that for Markov renewal processes event rates and time probabilities are insensitive to distributions of lifetimes. It should, thus, be clear why one can directly obtain the formulas above without needing to go through the analysis of the Markov chain in [3], and also why the results are insensitive to the backoff distribution.

IV. Fixed-Point Equation

Focusing on the backoff and attempt process of a node, and being given the collision probability γ the attempt rate is provided by $G(\gamma)$ in (1). It is important to recall that in the present discussion all rates are conditioned on being in the backoff periods. Later, we will see how to incorporate the channel activity periods. Now, if all nodes have the same backoff parameters, they will all see the same average collision probability, γ and, hence, will have the same attempt rate. If the attempt rate (or probability) of each node per slot is β, $0 \leq \beta \leq 1$; then, conditioning on an attempt of the given node, the probability of this attempt experiencing a collision is the probability that any of the other nodes attempts in the same slot. Under the decoupling approximation, the number of attempts made by the other nodes is binomially distributed with parameters β and $n-1$. Under the approximation, the number of attempts in successive slots form an i.i.d. sequence. The probability of collision of an attempt by a node is given by

$$\Gamma(\beta) := 1 - (1-\beta)^{(n-1)}. \quad (2)$$

We will show later in the paper that under a certain asymptotic regime the aggregate attempt rate $n\beta$ converges to a positive value as $n \to \infty$. Then (motivated by the binomial to Poisson convergence theorem) for a large number of nodes, it is reasonable to model the attempt process of the other nodes (with respect to a given node) as a sequence of i.i.d. batches (at slot boundaries) with the batch distribution being Poisson with mean $(n-1)\beta$. The collision probability under this model is then clearly given by

$$\Gamma(\beta) := 1 - e^{-(n-1)\beta}. \quad (3)$$

It is now natural to expect that the equilibrium behavior of the system will be characterized by the solutions of the following fixed-point equation

$$\gamma = \Gamma\left(G(\gamma)\right). \quad (4)$$

If this equation can be solved, it will yield the collision probability, from which the attempt rate can be obtained using (1). We will see in Section VI that throughputs can be obtained once these quantities are determined.

V. Analysis of the Fixed-Point Problem

Since $\Gamma(G(\gamma_k))$ is a composition of continuous functions it is continuous. We, thus, have a continuous mapping from [0,1] to [0,1]. Hence, by Brouwer's fixed-point theorem there exists a fixed point in [0,1]. We next turn to uniqueness.

Lemma 5.1: $G(\gamma)$ is nonincreasing in γ if b_k, $k \geq 0$, is a nondecreasing sequence.

Proof: Provided in the Appendix. ∎

Theorem 5.1: $\Gamma(G(\gamma))$: [0,1] $\to$ [0,1], has a unique fixed point if b_k, $k \geq 0$, is a nondecreasing sequence.

Proof: Since $\Gamma(\beta)$ is nondecreasing in β and, by Lemma 5.1, $G(\gamma)$ is nonincreasing in γ, it follows that $\Gamma(G(\gamma))$ is nonincreasing in γ. The fixed point must, therefore, be unique, since multiple fixed points will lead to a contradiction to the nonincreasing property of $\Gamma(G(\gamma))$. ∎

Remarks 5.1:

1) We observe that in the IEEE 802.11 standard the sequence b_k is nondecreasing. Hence, for the practical system there will be a unique fixed point.
2) In the above discussion, we have only considered *balanced* fixed points, i.e., ones in which all the nodes have the same value of collision probability γ. It is possible, however, under the decoupling approximation, to set up a system of fixed-point equations for *unbalanced* fixed points, i.e., ones in which the collision probability of node j is γ_j, with these

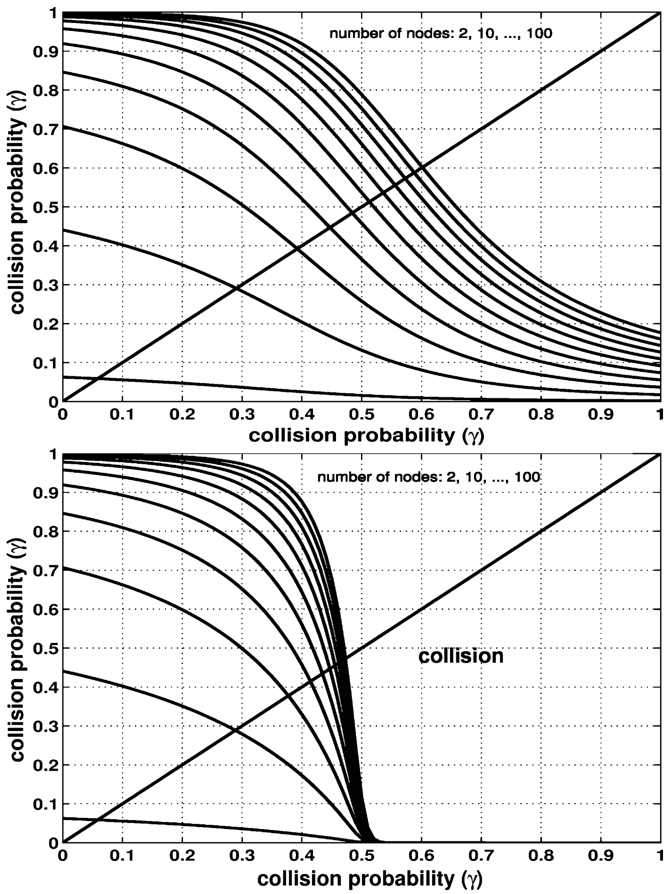

Fig. 5. Plots of $\Gamma(G(\gamma))$ versus γ for two values of K [(top) 7 and (bottom) 100], $b_0 = 16$ slots, and multiplicatively increasing b_k with multiplier $p = 2$. For each K, plots are shown for $n = 2, 10, 20, 30, 40, 50, 60, 70, 80, 90, 100$.

values being possibly different for different j. This yields the following set of equations:

$$\gamma_i = 1 - \prod_{j=1, j\neq i}^{n} (1 - G(\gamma_j))$$

for $1 \leq i \leq n$. By symmetry, we expect that the long run average operating point of the system will correspond to a balanced fixed point of these equations. However, in [9], we have shown that, in general, there can also exist unbalanced fixed points, which suggest *multistability*, and, indeed, simulations reveal that, in such cases, there is serious short-term unfairness. In [9], we also provide a sufficient condition for there to be no unbalanced fixed points. It turns out that the default IEEE 802.11 parameters satisfy these conditions. Thus, in practice, there will be a unique balanced fixed point and no unbalanced ones.

A. Examples and Comparison With ns2 Simulations

In Fig. 5, we show plots of $\Gamma(G(\gamma))$ versus γ for several parameters. Here, $p = 2$, as in the IEEE 802.11 standard. In the plot on the top, we use the value $K = 7$. In both of the plots, the initial mean backoff b_0 is 16 slots. The intersection of these plots with the "y = x" line corresponds to the fixed point. We see that the collision probability increases with an increasing number of nodes. For $n \geq 30$, with $K = 7$, the collision probability is larger than with $K = 100$. This is because with larger K, nodes are able to expand their backoff durations more and, hence, attempt less often. The collision probability for $n \leq 20$ is not sensitive to K for $K \geq 7$, since with $n \leq 20$ there are rarely more than seven consecutive collisions.

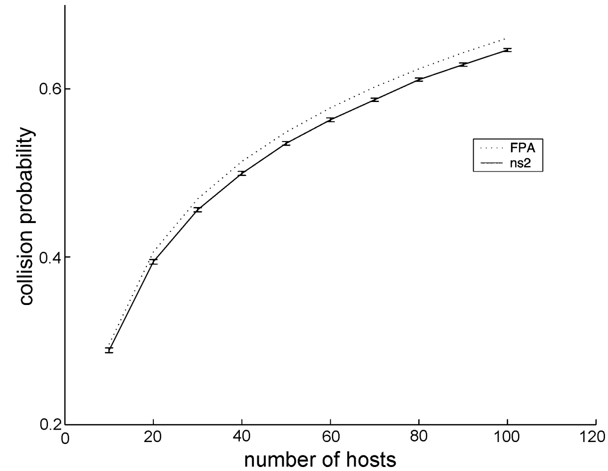

Fig. 6. Plot of collision probability versus number of nodes. Comparison of collision probability (γ) obtained from an *ns2* simulation (plot labeled *ns2*), and the fixed-point analysis (plot labeled *FP*). 95% confidence intervals are shown for the values obtained from the *ns2* simulation. In the *ns2* simulation the default IEEE 802.11 parameters are used: data rate: 11 Mb/s; control packet rate: 2 Mb/s.

It was reported in [3] that the fixed-point analysis works well for IEEE 802.11 parameters. In Fig. 6, we demonstrate this by plotting the collision probability obtained from the fixed-point method and from an *ns2* simulation.

In all the *ns2* simulations presented in this paper, we have used *ns2* version 2.26. The bugs present in the IEEE 802.11 code were patched by using an updated version of the code taken from the *ns2* snapshot dated January 5, 2004. Static routing was implemented by using *NOAH* code (dated November 2003), downloaded from the web site of J. Widmer, EPFL, (http://icapeople.epfl.ch/widmer/uwb/ns-2/noah/index.html). As can be seen, the fixed-point analysis provides a good approximation for a wide range of values of the number nodes.

B. Comparison With the Coupled Backoff DTMC

It can be seen that, when the backoff durations are geometrically distributed, then the coupled evolution of the backoffs of the nodes, as shown in Fig. 2, is exactly modeled by a discrete time Markov chain (DTMC). Hence, if the decoupling approximation works well, it should be able to match the results obtained from this DTMC. We now turn to this question. We proceed with the following assumptions: 1) the number of nodes $n \geq 2$; 2) exponential backoff with multiplier $p > 1$, i.e., $b_k = p^k b_0$, $1 \leq k \leq K$; 3) backoff durations are geometrically distributed, or, equivalently (with the b_k expressed in number of slots), when a node is in backoff stage k, it attempts in the next slot with probability $1/b_k$. We only need to consider the system backoff periods, and we index the slots in backoff time by $t = 0, 1, 2, \cdots$.

TABLE I
COLLISION PROBABILITIES: DTMC AND FPA; $K = 1$ AND $K = 2$, AND $b_0 = 16$

No. of Nodes	DTMC ($K=1$)	FPA ($K=1$)	DTMC ($K=2$)	FPA ($K=2$)
2	0.0598	0.0592	0.0595	0.0587
3	0.1111	0.1105	0.1088	0.1078
4	0.1568	0.1563	0.1510	0.1500
5	0.1983	0.1979	0.1879	0.1870
6	0.2365	0.2362	0.2209	0.2202
7	0.2720	0.2718	0.2508	0.2502
8	0.3052	0.3050	0.2782	0.2778
9	0.3363	0.3362	0.3036	0.3033
10	0.3657	0.3656	0.3272	0.3270
11	0.3933	0.3933	0.3494	0.3493
12	0.4196	0.4195	0.3703	0.3702
13	0.4444	0.4444	0.3900	0.3900
14	0.4680	0.4680	0.4088	0.4088
15	0.4905	0.4905	0.4266	0.4266
16	0.5119	0.5119	0.4436	0.4436
17	0.5323	0.5323	0.4598	0.4599
18	0.5518	0.5518	0.4754	0.4755
19	0.5703	0.5703	0.4903	0.4904
20	0.5881	0.5881	0.5046	0.5048

It is convenient to work with the process that counts the number of nodes in each backoff stage. This will be a $(K+1)$-dimensional process for any number of nodes. Define the number of nodes in the backoff stage $k \in \{0, 1, \cdots, K\}$ in slot t to be $M_k^{(n)}(t)$. Let $\mathbf{M}^{(n)}(t)$ denote the vector random process with components $M_k^{(n)}(t)$. From the foregoing, it is clear that $\mathbf{M}^{(n)}(t)$ is a Markov process taking values in the set $\mathcal{M}^{(n)} := \{\mathbf{m} : m_k \text{ nonnegative integers}; \sum_{k=0}^{K} m_k = n\}$.

Theorem 5.2: [10] For $b_0 > 1$, and $p > 1$, the DTMC $\mathbf{M}^{(n)}(t)$ on $\mathcal{M}^{(n)}$ is irreducible. ■

It follows that, under the conditions $b_0 > 1$ and $p > 1$, the DTMC $\mathbf{M}^{(n)}(t)$ is positive recurrent. Let $\boldsymbol{\pi}^{(n)}$ denote the stationary probability measure on $\mathcal{M}^{(n)}$.

For small values of K (e.g., 1 or 2), $\boldsymbol{\pi}^{(n)}$ can be numerically computed. Now given $\boldsymbol{\pi}^{(n)}$, the collision probability γ can be obtained in a straightforward manner (see [5] and [10]). Sample results are shown in Table I. Results are shown for $K = 1$ and $K = 2$, and $b_0 = 16$. It can be seen that the fixed-point analysis approximates the collision probability very well.

VI. CALCULATING THROUGHPUTS

We make two key observations. The first is demonstrated by Fig. 7. Because of the i.i.d. batch binomial assumption on the aggregate attempt process, the instants at which a successful transmission or a collision ends are renewal instants. Each such instant is followed by a time until the next attempt, followed by a collision or a success, and so on. The second observation is that since all the nodes follow the same backoff process, each node has an equal probability of winning the allocation "race." With this in mind, we can now discard the backoff times and focus only on the times when an attempt is made and on the intervening channel activity. A successful attempt leads to the channel being allocated to one of the n contending nodes with equal probability. Hence, in a saturated system, in order to compute the amount of time the channel will be allocated to a node, we only need to know the identity of the packet that will be found at the head-of-the-line if the channel is allocated to the node.

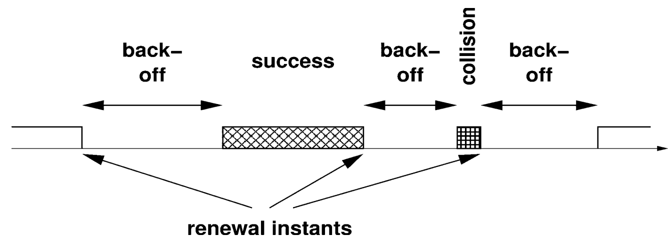

Fig. 7. Aggregate process of backoffs and channel activity.

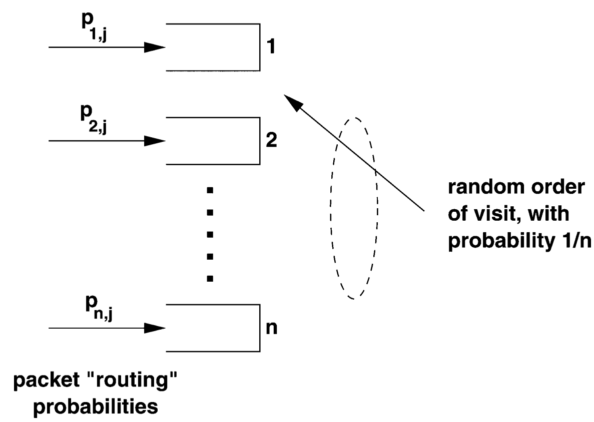

Fig. 8. n transmitters are served in random order with equal probability for each node.

Consider the model shown in Fig. 8. The nodes are visited in random order with equal probability. Each node receives an *open loop* stream of packets. There are m_i streams being handled by node i. These are indexed by $1 \le j \le m_i$; these would represent m_i flows from node i to some of the other nodes. We can, thus, use the term "flow (i,j)."

By "open-loop," we mean that packets arrive to the node and have to be delivered; there are no acknowledgement and flow control as in TCP-controlled traffic. A fraction $p_{i,j}$ of the packets at node i belong to stream j, $1 \le j \le m_i$. Since the node is saturated, there is always a packet at the head-of-the-line when the channel is allocated to any node, and $p_{i,j}$ is the probability that the packet is from flow j. Let us define the packet length of flow (i,j) to be $L_{i,j}$ and the physical transmission rate for flow (i,j) to be $C_{i,j}$ *bits per slot*.

In addition, we define the following.

$T_o :=$ Fixed overhead with a packet transmission in slots (e.g., IEEE 802.11b: $T_o = 52$ slots).

$T_c :=$ Fixed overhead for an RTS collision in slots (e.g., IEEE 802.11b: $T_c = 20$ slots).

The above two observations, the traffic model described above, and the parameters listed above lead to the expression in (5) for the saturation throughput of flow (i,j) (in bits per slot) given the collision probability γ and the per node attempt rate β [see (5), shown at the bottom of the next page].

The formula follows from the renewal reward theorem. The mean renewal time (see Fig. 7) is the mean time until an attempt, plus the mean time for channel activity; i.e., a transmission or a collision. The mean time until an attempt is $1/(1-(1-\beta)^n)$, which assumes that the aggregate attempt process is binomial.

When there is an attempt the channel is allocated to node i (with probability $(\beta(1-\beta)^{n-1})/(1-(1-\beta)^n)$), else there is a collision, for which the channel will be busy for the time T_c. If the channel reservation succeeds, then the head-of-the-line packet at node i is of flow (i,j) with probability $p_{i,j}$, and transmitting this takes the time $(L_{i,j}/C_{i,j})+T_o$. The mean reward during the cycle is $((\beta(1-\beta)^{n-1})/(1-(1-\beta)^n))p_{i,j}L_{i,j}$. These terms when put together, using the renewal-reward theorem, yield the displayed expression in (5) (after canceling the term $1-(1-\beta)^n$).

A. Low-Speed Transmitters Bound all Throughputs

It has been observed (see, for example, [2]) that when there are several flows with different physical transmission rates, then the throughput of all the flows is bounded by the slowest transmission rate. We can examine this observation using (5).

If two nodes i_1 and i_2 are such that for some j_1, $1 \le j_1 \le m_1$, and j_2, $1 \le j_2 \le m_2$, $p_{i_1,j_1}L_{i_1,j_1} = p_{i_2,j_2}L_{i_2,j_2}$, then it follows from (5) that $\theta_{i_1,j_1}(\gamma,\beta) \le \min\{C_{i_1,j_1}, C_{i_2,j_2}\}$ and $\theta_{i_2,j_2}(\gamma,\beta) \le \min\{C_{i_1,j_1}, C_{i_2,j_2}\}$, i.e., the flow with the lower physical rate will bound the throughput of both.

Remark: The above analysis points to an important observation. Suppose we are interested in achieving flow throughputs that are proportional to their physical link rates; i.e., $\theta_{i,j} = \nu C_{i,j}$ for some ν. It has been suggested in previous literature that this can be achieved by appropriately choosing the packet lengths. We notice from (5) that the desired throughput proportionality can be achieved only by making $L_{i,j}$ proportional to $C_{i,j}/p_{i,j}$, which requires knowledge of the $p_{i,j}$s, which may not be practicable.

Let us now consider a simpler situation with n nodes each being the transmitter for a single flow and all packet lengths being equal to L. Then the *total* network throughput is given by (6), shown at the bottom of the page. Since the denominator is bounded below by $\sum_{i=1}^{n}(\beta(1-\beta)^{n-1}((L/C_i)+T_o))$, it can be seen that

$$\Theta(\beta) \le \frac{1}{\frac{1}{n}\sum_{i=1}^{n}\frac{1}{C_i}} \le n \times \min_{1\le i\le n} C_i$$

i.e., *the total network throughput is bounded above by the reciprocal of the harmonic means of the physical bit rates of the n flows*. Thus, for example, if there are two flows with physical rates 2 and 4 Mb/s, then the total network throughput will be bounded by $(2/(1/2)+(1/4)) = 2.667$ Mb/s. Also, with equal packet lengths, we see that this total throughput is shared equally among all the flows.

VII. Asymptotic Analysis

If we numerically examine the fixed points (see [10]), we notice that the fixed points appear to be converging as K becomes large, and there is not much variation in them for $K \ge 15$. Thus, we are motivated to analyze the fixed point for $K \to \infty$. A similar asymptotic analysis has also been carried out independently by Kwak *et al.* in [7]; while their final results are the same as our Theorem 7.2, we have displayed an analytical form for the fixed-point solution (see Theorem 7.1), and we derive our asymptotic results by taking a limit in this solution. Further, we also provide a relaxed fixed-point iteration for computing the fixed point (see Section VII-A).

To permit closed form analysis, let us take $b_0 = b$ slots, and $b_k = p^k \times b_0$, where $p \ge 1$; hence, by Theorem 5.1, a unique fixed point still exists. The multiplicative increase is in any case a part of the IEEE 802.11 standard; we are generalizing to an arbitrary multiplier in order to study the impact of the value of this multiplier.

Assuming $\gamma < 1/p$, and taking $K \to \infty$, we see that

$$G(\gamma) = \frac{1}{b_o} \times \frac{1-p\gamma}{1-\gamma}.$$

Note that the assumption that $\gamma < 1/p$ does not affect the fixed-point analysis presented earlier, since we will see in Theorem 7.2 that the fixed point in the limit $K \to \infty$ is less than $1/p$.

Given γ, $G(\gamma)$ is the probability of attempt of any node. Then using the batch Poisson version of the collision probability in (3), the fixed-point equation becomes

$$\gamma = f(\gamma) \text{ where } f(\gamma) := 1-\exp\left(-\frac{n-1}{b_0} \times \frac{1-p\gamma}{1-\gamma}\right). \quad (7)$$

In order to obtain compact expressions, let us define $\eta = (n-1)/b_0$.

$$\theta_{i,j}(\beta) = \frac{\beta(1-\beta)^{n-1}p_{i,j}L_{i,j}}{1+\sum_{i=1}^{n}\left(\beta(1-\beta)^{n-1}\left(\left(\sum_{k=1}^{m_i}p_{i,k}\frac{L_{i,k}}{C_{i,k}}\right)+T_o\right)\right)+\left(\left(1-(1-\beta)^n-n\beta(1-\beta)^{n-1}\right)T_c\right)} \quad (5)$$

$$\Theta(\beta) = \frac{n\beta(1-\beta)^{n-1}L}{1+\left[\sum_{i=1}^{n}\beta(1-\beta)^{n-1}\left(\frac{L}{C_i}+T_o\right)\right]+\left(\left(1-(1-\beta)^n-n\beta(1-\beta)^{n-1}\right)T_c\right)} \quad (6)$$

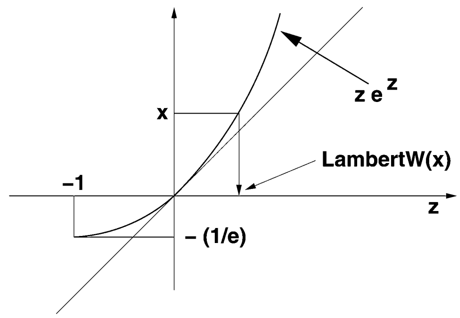

Fig. 9. LambertW function is the inverse function of ze^z; notice that, for $x \geq -(1/e)$, $\mathrm{LambertW}(x) \leq x$, with equality only for $x = 0$.

Theorem 7.1: The fixed point is of the form

$$\gamma(\eta) = \frac{\mathrm{LambertW}\left(\eta(p-1)e^{\eta p}\right) - \eta(p-1)}{\mathrm{LambertW}\left(\eta(p-1)e^{\eta p}\right)}.$$

Remark: For $x \geq -1/e$, $\mathrm{LambertW}(x)$ is defined as the inverse of the function ze^z (see Fig. 9).

Proof: We proceed from (7). Writing $\nu = 1 - \gamma$, and using the definition of η, this equation can be rewritten as $\nu = \exp(-\eta p)\exp(\eta(p-1)/\nu)$. Multiplying both sides by $\eta(p-1)$, we obtain $\eta(p-1)\exp(\eta p) = (\eta(p-1)/\nu)\exp(\eta(p-1)/\nu)$. It follows from the definition of $\mathrm{LambertW}(\cdot)$ that $(\eta(p-1)/\nu) = \mathrm{LambertW}(\eta(p-1)\exp(\eta p))$ from which the result follows by substituting $1 - \gamma$ for ν. ■

A. Relaxed Fixed-Point Iteration

The fixed-point $\gamma(\eta)$ can only be computed numerically. In this section, we provide a relaxed fixed-point iteration. With reference to (7), and, with $\gamma_0 := 1/p$, consider the sequence of values generated by the iterations

$$\gamma_{k+1} = (1-\alpha)f(\gamma_k) + \alpha\gamma_k \tag{8}$$

where $0 < \alpha < 1$. Notice that $\alpha = 0$ corresponds to the usual fixed-point iteration, which will converge if $f(\gamma)$ is a contraction. The above iteration is called a *relaxed* fixed-point iteration. We will now provide a condition on α that will ensure that the iterates converge to the fixed point.

First of all, since $f(\gamma)$ is continuous, it is clear from the iteration in (8) that if the sequence of iterates converge then they must converge to the fixed point. It is also clear that if, for each k, $\gamma_k \geq f(\gamma_k)$, then the sequence $\{\gamma_k\}$ is nonincreasing. This follows because $\gamma_{k+1} = (1-\alpha)f(\gamma_k) + \alpha\gamma_k \leq \gamma_k$ if and only if $f(\gamma_k) \leq \gamma_k$. Thus, since $\gamma_k \geq 0$, for the convergence of the sequence $\{\gamma_k\}$ it suffices to ensure that $\gamma_k \geq f(\gamma_k)$ for all k.

Now, it can easily be shown that the derivative of $f(\gamma)$ at $\gamma = 1/p$ is given by $D := -((n-1)/b_0)(p^2/p - 1)$, and that, for $\gamma_{k+1} \leq \gamma_k$

$$f(\gamma_{k+1}) - f(\gamma_k) \leq |D|(\gamma_k - \gamma_{k+1}).$$

From this inequality, we can see that to ensure $f(\gamma_k) \leq \gamma_k$, for all k, it is sufficient to ensure that, for all k, $|D|(\gamma_k - \gamma_{k+1}) \leq \gamma_{k+1} - f(\gamma_k)$. Using the iteration in (8), this is equivalent to ensuring that, for all k, $|D|(1-\alpha)(\gamma_k - f(\gamma_k)) \leq \alpha(\gamma_k - f(\gamma_k))$. Hence, it suffices that $|D|(1-\alpha) \leq \alpha$, or that $\alpha \geq |D|/(|D|+1)$. Thus, for example, with $n = 10$ nodes, $b_0 = 16$ slots, and $p = 2$, the relaxed fixed-point iteration with α such that $(2.25/3.25) < \alpha < 1$ will yield the unique fixed-point $\gamma(\eta)$.

B. Taking n to ∞

We now wish to take n to ∞ and study the limit of the fixed-point solution obtained in Theorem 7.1. For this, we need the following properties of the LambertW function.

Lemma 7.1:

1) For $a > 0$

$$\lim_{x\to\infty} \frac{\mathrm{LambertW}(axe^x)}{x} = 1. \tag{9}$$

2) For $a > 0$

$$\lim_{x\to\infty} \left(\mathrm{LambertW}(axe^x) - x\right) = \ln a.$$

3) For $0 < a \leq 1$ $\mathrm{LambertW}(axe^x) \leq x$.
4) For $0 < a < 1$, the convergence in (9) is from the following.

Proof: Provided in the Appendix. ■

The following result is now obtained by applying Lemma 7.1 to the expression for γ in Theorem 7.1.

Theorem 7.2:

1) $\gamma(\eta) < 1/p$.
2) $\lim_{n\to\infty} \gamma(\eta) \uparrow 1/p$.
3) $\lim_{n\to\infty} n\beta \uparrow \ln(p/p-1)$. ■

The following result provides the rate of convergence in Theorem 7.2.

Theorem 7.3: We have

$$\lim_{n\to\infty} \eta p \left(\gamma(\eta) - \frac{1}{p}\right) = a\ln(a) \text{ where } a = \frac{p-1}{p}.$$

Proof: Define $x = \eta p$. From Theorem 7.1, we have

$$\begin{aligned}\gamma(\eta) - \frac{1}{p} &= 1 - \frac{x}{\mathrm{LambertW}(axe^x)}a - \frac{1}{p} \\ &= a\left(1 - \frac{x}{\mathrm{LambertW}(axe^x)}\right).\end{aligned}$$

Thus, we obtain

$$\eta p\left(\gamma(\eta) - \frac{1}{p}\right) = a\frac{x}{\mathrm{LambertW}(axe^x)}\left(\mathrm{LambertW}(axe^x) - x\right).$$

Now using Parts 1 and 2 of Lemma 7.1, we obtain the desired conclusion. ■

Remarks 7.1:

1) Theorem 7.2 provides explicit expressions for the collision probability and the fixed point for large K and a large number of nodes. We see that for large n the collision probability is directly related to the backoff multiplier p, and is the reciprocal of this multiplier.
2) We also see that $n\beta$, the mean attempt rate per slot, goes to $\ln(p/p-1)$ and, hence, the attempt probability per node (during backoff periods) behaves like $O(1/n)$. This lends some support to the original assumption that from the point of view of a node the attempt process of the other nodes can be viewed as an independent process with i.i.d. batch Poisson arrivals in successive slots.

C. Asymptotic Aggregate Throughput

Let us now consider n nodes handling n flows with all the flows having the same transmission rate, C. The aggregate throughput of the network is given by [compared with (6)]

$$\Theta(\beta)=\frac{n\beta e^{-n\beta}L}{1+\left(n\beta e^{-n\beta}\left(\frac{L}{C}+T_o\right)\right)+((1-e^{-n\beta}-n\beta e^{-n\beta})T_c)}.$$

We infer from this equation that, as $n \to \infty$, the aggregate throughput converges to

$$\tau(p):=\frac{(1-\frac{1}{p})L}{\frac{1}{\ln\left(\frac{p}{p-1}\right)}+\left(1-\frac{1}{p}\right)\left(\frac{L}{C}+T_o\right)+\left(\frac{\frac{1}{p}}{\ln\left(\frac{p}{p-1}\right)}-\left(1-\frac{1}{p}\right)\right)T_c}.$$

The following result is then immediately obtained.

Theorem 7.4:
1) $\lim_{p\to\infty}\tau(p)=0$.
2) $\lim_{p\to 1}\tau(p)=0$.
3) $\tau(p)$ is maximized at

$$p=\frac{\frac{T_c}{T_c+1}}{\mathrm{LambertW}\left(-\frac{1}{e}\cdot\frac{T_c}{(T_c+1)}\right)+\frac{T_c}{T_c+1}}.$$

■

Remarks 7.2:
1) The behavior of the aggregate throughput as p goes to its two extremes is as expected. If $p \to 1$, then the nodes do not increase their backoff intervals in response to collisions. The collision probability becomes large and the throughput drops to 0. Obviously, as $p \to \infty$, collisions cause a drastic reduction in attempts essentially shutting the nodes off.
2) In an attempt to see what the above asymptotic results have to say about realistic network parameters, in Fig. 10, we plot the aggregate throughput for finite K and finite n, using the formula in (6) with equal transmission rate for all the flows. We see that the throughput increases steeply for $1 < p < 2$, but is quite flat with p after $p = 2$. There is an optimal value of p, but unless p is very close to 1, the throughput is not very sensitive to p. It can be seen that the backoff multiplier used in the standard, i.e., $p = 2$, is adequate unless the number of nodes becomes very large. For $T_c = 17$ (slots), the third part of Theorem 7.4 returns $p = 3.85$, which compares well with the curve for $n = 60$ in Fig. 10.

VIII. Application to the Analysis of TCP-Controlled File Transfers

A. Some Modeling Assumptions

We will make the following assumptions.

A1: The files are infinitely long. Thus, we do not deal with web transfers. Practically, this assumption means that our analysis applies to large file transfers, such as software, document, or media downloads.

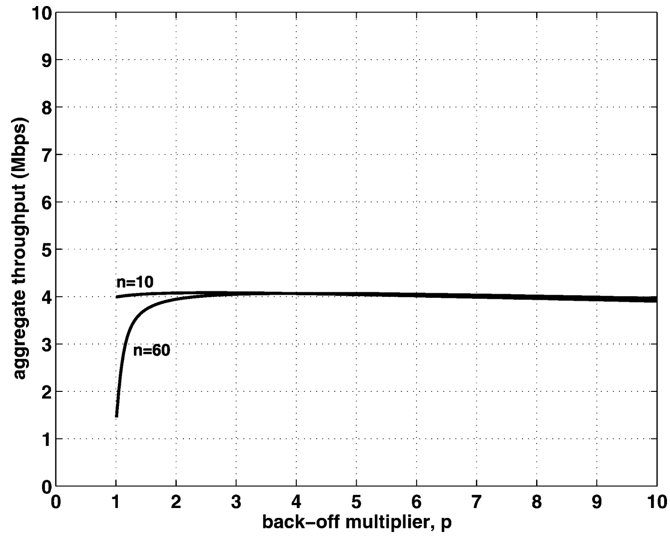

Fig. 10. Aggregate throughput plotted versus the backoff multiplier p for two values values of n. The network parameters are $K = 10$, $b_0 = 16$ slots, data packet length 8 000 bits, packet overhead 592 bits, slot time 20 μs, transmission rate for all flows 11 Mb/s, fixed (rate independent) data packet transmission overhead 52 slots, collision overhead 17 slots.

A2: The modulation scheme and bit rate of the physical connection between a pair of communicating wireless devices is ideally adapted (but fixed) so that there is no packet loss owing to bit errors. Further, the retransmission time-out at each TCP transmitter is large enough so that time-outs never takes place.

A3: At the transmitter of each wireless device, the capacity of the buffer is such that there is no packet loss. This assumption effectively holds in practice if the number of file transfer connections through a node is small enough so that the sum of the maximum TCP windows of all the connections is less than the buffer size. For, say, ten connections, this would typically require a buffer of no more than 512 KB.

A4: The file transfer throughputs are bottlenecked only by the rates they obtain over the WLAN. For example, the transfers could be between the wireless devices across an ad hoc WLAN, or, in the infrastructure case, between the wireless devices and devices attached to a high-speed wired LAN to which the AP is attached. For transfers within a building or campus, this assumption is practically valid since most wired LANs are based on 100-Mb/s to 1-Gb/s Ethernet.

Owing to Assumption A1 it makes sense to talk about the long run time average throughput of a transfer. From Assumptions A2 and A3, it follows that the TCP window of each connection grows to its maximum value, and by Assumption A4, each data packet or ACK of all the TCP connections will be queued at the transmitter of one of the WLAN devices.

Let us adopt the following connection model. There are m connections, indexed by j, $1 \le j \le m$. The source node of connection j is denoted by $s(j)$, and the receiver node is denoted by $r(j)(\neq s(j))$. Thus, for connection j, the TCP ACKs will queue up at the transmitter of node $r(j)$. The data packet length for connection j is denoted by L_j and the ACK packet length by $L_j^{(\mathrm{ack})}$. In general, each node will transmit data packets for some connections and ACK packets for other connections.

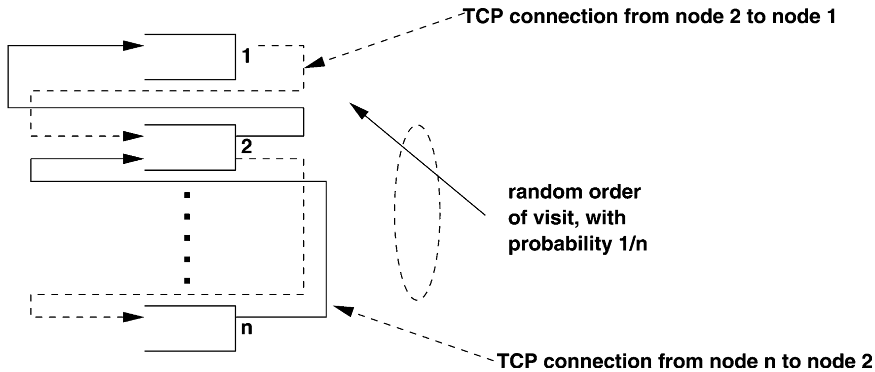

Fig. 11. There are several TCP connections, modeled as "chains" of customers with a fixed population (the window size) circulating in a random polling network. The solid arrows between the queues show the direction of TCP data transfer for a connection, and the dashed arrows show the direction of TCP ACK transmission. The n transmitters are served in random order with equal probability for each node.

In order to use the "saturated queues" analysis presented earlier in the paper, we make the following additional assumption

A5: The configuration of the TCP connections and the sizes of their windows are such that the transmitter queues of the wireless devices never empty out.

Remark: This assumption is made to permit us to use the fixed-point analysis presented earlier in the paper. It, however, considerably restricts the scenarios to which the analysis will apply. For example, the common situation of two or more devices simultaneously downloading files via an AP is not covered by our analysis. This is because the AP needs to send many more packets for each packet that each of the devices sends, and, hence, the device queues will empty out, violating our saturated queues assumption.

We will utilize Assumption A5 as follows. Recall our discussion in Section VI. If all the n queues always have packets to send, then they always contend for the channel, and each successful attempt ''belongs'' to each of the queues with equal probability, $1/n$.

B. A Formula for Connection Throughput

Let us now focus only on the successful attempt instants. Such a success belongs to node i with probability $1/n$. The HOL packet at that node is then transmitted. If this packet is of length L and the transmission rate is C, then a time $(L/C)+T_o$ elapses. If the packet transmitted is a data packet, then possibly an ACK is inserted into the transmitter queue of the receiving node (note that if delayed ACKs are used then not every data packet causes an ACK to be generated). On the other hand, if the packet transmitted is an ACK packet, then one or more packets are inserted into the transmitter queue of the receiving node. *Thus, the queues can be viewed as evolving only at successful polling instants*. This is an important observation as it allows us to ignore the backoff periods while analyzing the evolution of the packet queues. Note that this observation does not hold if there are finite rate open-loop arrival processes into the nodes, as these arrival processes will cause the queues to evolve even during backoff periods.

From the above observations, we can now proceed by analysing the discrete time random polling model shown in Fig. 11. The discrete "time" in this model evolves over packets. Note that we do not need to be concerned with packet lengths (data or ACK), or physical bit rates. We will see that all we need from this model is the fraction of polls to a queue that find packets of each type at the head-of-the-line. There are several TCP connections modeled as "chains" or classes of customers circulating between pairs of nodes. The populations of the chains are the TCP window sizes. If the delayed ACK parameter for a connection is greater than 1 (let us say 2), then at the receiving node for that connection, two data packets give rise to one ACK packet. We can view this ACK packet as being a batch of 2 that is served together.

The state of the random polling model is the position and type of each packet in each queue. This process evolves over packet times. It is easy to see that the evolution of this rather complicated process is Markovian. Analysis of this Markov chain will yield the following probabilities, that will be used in the throughput formulas.

$h_{i,j} :=$ The probability that, at a polling instant, the HOL packet at node i is a data packet from connection j.

$h_{i,j}^{(\mathrm{ack})} :=$ The probability that, at a polling instant, the HOL packet at node i is an ACK packet for connection j (for which node i is the receiver node, i.e., $i = r(j)$)

By the observations made just before these definitions, we can conclude that *the probabilities* $h_{i,j}$ and $h_{i,j}^{(\mathrm{ack})}$ *do not depend on data and ACK packet lengths, nor on the physical bit rates of the connections*. These probabilities will depend only on the maximum TCP window sizes, the delayed ACK thresholds, and the connection configurations (i.e., which nodes carry which connections). We also note that once we have these probabilities, the throughput of connection j can be immediately obtained as in (10) [see also (5)], where (γ, β) are obtained from the fixed-point analysis. This formula has the same form as the one in (5). In the numerator the term $\beta(1-\beta)^{n-1}$ is the probability that node $s(j)$ has a success, $h_{s(j),j}$ is the probability that the HOL packet belongs to connection j, and when both these events occur connection j has a "reward" of L_j bits. The denominator is the mean length of a backoff and attempt cycle [see (10), shown at the bottom of the next page].

Remarks 8.1:
1) To be technically correct, (10) should have been obtained as the ratio of two expectations with respect to the stationary distribution of the Markov chain describing the random polling model. We have shown only the final result in terms of the HOL probabilities at the polling instants, as this is simple and intuitively clear.
2) In (5), the HOL probabilities were obtained from the ratios of the open-loop arrival rates into the queues. In (10), however, the HOL probabilities will need to be obtained from the packet level analysis of the random polling model shown in Fig. 11. We will show how this is done in the next subsection.
3) The denominator of the expression now includes a term for the service provided to TCP ACKs.
4) We have used the fact that all data packets within TCP connection j have the same length L_j, and the ACK packets within TCP connection j have the same size $L_j^{(\mathrm{ack})}$. If this were not the case, then we would need to make a more elaborate definition of the HOL probabilities which would have to include the probability of finding packets of each possible length.

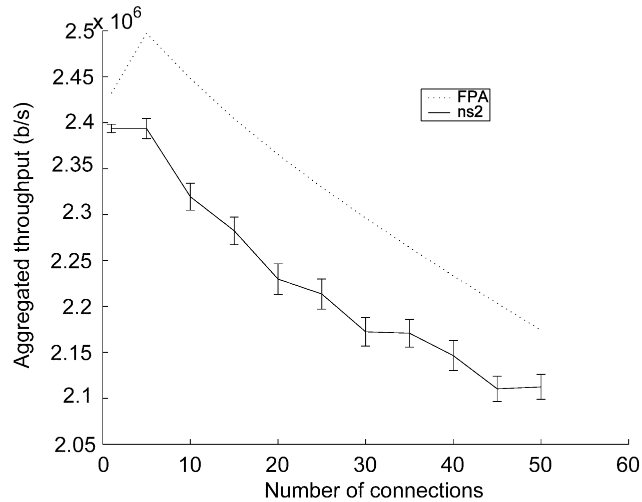

Fig. 12. Plot of aggregate TCP-controlled file transfer throughput versus number of simultaneous transfers over an IEEE 802.11b network. Results obtained from the approximate analysis, and from *ns2* simulations are shown, with 95% confidence intervals. In the simulation the IEEE 802.11 parameters are used, with data rate: 11 Mb/s, control rate: 2 Mb/s.

C. Obtaining the HOL Probabilities

Let λ_j be the throughput of connection j through its sender node $s(j)$ in the random polling model shown in Fig. 11. Thus, λ_j the average number of packets of connection j that pass through the node $s(j)$ per packet served in the polling model.

Theorem 8.1: If at each success instant one of the nodes is polled with equal probability (i.e., we have the model in Fig. 11), then $h_{s(j),j} = \lambda_j n$.

Proof: Let $\pi_{s(j),j}$ denote the fraction of packet services in the model of Fig. 11 during which the HOL position at node $s(j)$ is occupied by a data packet of connection j. Since the mean time that a packet spends in the HOL position is n, by Little's Theorem we have $\pi_{s(j),j} = \lambda_j n$. Owing to random polling, the HOL position at node $s(j)$ is observed by a Bernoulli process with probability of "success" equal to $1/n$. Hence, by the result that Bernoulli "arrivals" see time averages, we can conclude that $h_{s(j),j} = \pi_{s(j),j} = \lambda_j n$. ■

Remarks 8.2:
1) If the throughput of ACKs for connection j through its receiver node $r(j)$ is $\lambda_j^{(\mathrm{ack})}$, then, by the same argument as in Theorem 8.1, it follows that $h_{s(j),j}^{(\mathrm{ack})} = \lambda_j^{(\mathrm{ack})} n$.
2) We note that the hypothesis of the Theorem 8.1 that "at each success instant one of the nodes is polled with equal probability" requires the saturation assumption, i.e., Assumption A5, to hold. There are TCP connection configurations for which this assumption will not hold. For example, consider a single TCP connection from Node 1 to Node 2. The TCP receiver uses a delayed ACK threshold of 2; i.e., it returns one ACK for two received data packets. Clearly, over a large number of packets transmitted, we cannot say that about half will come from Node 1 and the other half from Node 2. In this case, the receiver node will tend to empty out and the saturation assumption will not apply. On the other hand, if Node 2 was also sending to Node 1, then our analysis will apply.
3) In view of Theorem 8.1, we need to analyze the random polling model and obtain the λ_js and the $\lambda_j^{(\mathrm{ack})}$s, and this will yield the HOL probabilities needed in the throughput formula.

D. Comparison With ns2 Simulations

In Fig. 12, we compare the results from the analysis presented above and *ns2* simulations; 95% confidence intervals are shown around the simulation results. For comments on the version of *ns2* used, see Section V-A. The scenario simulated is that there are n nodes paired up with n other nodes; each node in the first group is performing a TCP-controlled long file transfer to its corresponding member in the other group. The maximum receiver window for each TCP connection is 20 packets, the TCP packet length is 1 KB, and the receivers do not delay the ACKs (i.e., an ACK is returned for each received data packet). In this situation, of course, $h_{i,j}$ will be 1 whenever Node i is the source node of connection j, and $h_{i,j}^{\mathrm{ack}}$ will be 1 whenever Node i

$$\theta_j(\beta) = \frac{\beta(1-\beta)^{n-1} h_{s(j),j} L_j}{1+\sum_{i=1}^{n} \beta(1-\beta)^{n-1}\left(\left(\sum_{\{j:s(j)=i\}} h_{i,j} \frac{L_j}{C_{i,r(j)}} + \sum_{\{j:r(j)=i\}} h_{i,j}^{(\mathrm{ack})} \frac{L_j^{(\mathrm{ack})}}{C_{i,s(j)}}\right) + T_o\right) + \left(1-(1-\beta)^n - n\beta(1-\beta)^{n-1}\right) T_c} \tag{10}$$

is the receiver node for connection j. The physical link rates are all 11 Mb/s. The aggregate throughput over all the connections is plotted versus the number of connections. We notice that the match between analysis and simulations is good, with the worst case error being about 6%. The simulation trace file showed that during the simulations there was no TCP time-out; thus, our Assumption A2 held in this case.

Another scenario that we evaluated was two nodes sending files to each other, simultaneously. In this case, the aggregate throughput predicted by the model is 2.4164 Mb/s, while *ns2* simulations return a 95% confidence interval of $[2.4054, 2.4153]$ Mb/s. In the simulation, the throughput obtained by each transfer is approximately a half of the aggregate throughput.

IX. Summary

Our analysis has provided a simple and general representation of the fixed-point equation that arises from an analysis initiated by Bianchi in [3]. The representation is insensitive to the distribution of the backoff times. We show that, if the mean backoff durations for successive retrials are monotone nondecreasing, then the fixed-point equation has a unique solution. Then we provide general throughput formulas for open-loop arrival processes (e.g., UDP transfers). We recover the observation that connections with small physical rates dominate the throughputs of other connections. We then turn to the special case of exponential backoff with an arbitrary positive multiplier, p, and where we do not limit the number of retrials a node can make. This leads to simpler expressions which permit us to study the network performance as the number of nodes goes to infinity. For this case, we obtain a characterization of the fixed-point solution for the collision probability for each n. Then we take n to ∞ and obtain the limit of the collision probability and aggregate attempt rate that agree with the results of Kwak *et al.* in [7]. We also provide a relaxed fixed-point iteration for computing the fixed point for any finite n when the number of retrials is not limited. The asymptotic aggregate throughput is obtained and from this the optimal backoff multiplier p is also derived.

For exponential backoff, and geometrically distributed backoff periods, the backoff process can be modeled via a discrete time Markov chain. In Section V-B, we study this DTMC, and for some simple computable cases, we compare the collision probability obtained from the DTMC with that obtained from the fixed-point analysis.

Finally, we show how the saturation throughput analysis can be used to obtain TCP-controlled file transfer throughputs for some network scenarios. In this analysis we exploited the idea that, for window-controlled traffic, the backoff process evolution can be decoupled from the packet service process, the latter being modeled by a random polling queue.

Appendix

Proof: (Lemma 5.1) We have

$$G(\gamma) := \frac{1+\gamma+\gamma^2+\cdots+\gamma^K}{b_0+\gamma b_1+\gamma^2 b_2+\cdots+\gamma^k b_k+\cdots+\gamma^K b_K}$$

and we need to show that the derivative of this function with respect to γ is negative. Taking the derivative we find that we need to show that

$$\sum_{k=0}^{K} b_k\gamma^k\left(\sum_{j=1}^{K} j\gamma^{(j-1)}\right) \le \sum_{k=0}^{K}\gamma^k\left(\sum_{j=1}^{K} jb_j\gamma^{(j-1)}\right)$$

i.e.,

$$\sum_{k=0}^{K}\sum_{j=1}^{K} jb_k\gamma^{(k+j-1)} \le \sum_{k=0}^{K}\sum_{j=1}^{K} jb_j\gamma^{(k+j-1)}$$

or, equivalently, we need to show that

$$\sum_{n=1}^{2K}\gamma^{(n-1)}\sum_{\substack{j=\max\{(n-K),1\}\\ k=(n-j)}}^{\min\{n,K\}} j(b_j-b_k)\ge 0.$$

Now we consider each term $\sum_{\substack{j=\max\{(n-K),1\}\\ k=(n-j)}}^{\min\{n,K\}} j(b_j-b_k)$ and show that it is nonnegative. To this end, define

$$m(n)=|\{(j,k): j+k=n, 1\le j\le K, 0\le k\le K\}|$$

where $|\cdot|$ denotes set cardinality. When $k=j, jb_j - jb_k = 0$ and the corresponding term vanishes from the sum. Also, k equals 0 only when $j=n$ and $1\le n\le K$. Hence, simplifying the above expression, we get

$$\sum_{j=\max\{(n-K),1\}}^{\max\{(n-K),1\}+\left\lfloor\frac{m}{2}\right\rfloor-1} \left(((n-j)-j)\left(b_{n-j}-b_j\right) + n(b_n-b_0)1_{\{1\le n\le K\}}\right)$$

which is nonnegative since, in the range of the sum, $(n-j)-j\ge 0$ and $b_{n-j}-b_j\ge 0$. It is also easily seen that the derivative of $G(\cdot)$ is strictly negative for $\gamma>0$ if the b_k are not all equal, this implies that $G(\cdot)$ is strictly decreasing in this case. ■

Proof: (Theorem 7.1) This is just a simple manipulation of the fixed-point equation to get it into the form of the LambertW function. The fixed-point equation is

$$\gamma = 1-e^{\left(-(n-1)\times\frac{1}{b_0}\times\frac{1-p\gamma}{1-\gamma}\right)}$$

which can be rewritten as $(1-\gamma)=e^{-\eta p}e^{(\eta(p-1)/(1-\gamma))}$. This expression can be rearranged as follows:

$$\eta(p-1)e^{\eta p} = \frac{\eta(p-1)}{(1-\gamma)}e^{\frac{\eta(p-1)}{(1-\gamma)}}.$$

It follows, from the definition of the LambertW function (and utilizing the fact that $p>1$) that

$$\frac{\eta(p-1)}{(1-\gamma)} = \mathrm{LambertW}\left(\eta(p-1)e^{\eta p}\right).$$

The result follows by rearranging the equation to extract γ. ■

Proof: (Lemma 7.1)

1) For $x \geq 0$, write $z(x) = \mathrm{LambertW}(axe^x)$, i.e., $z(x)e^{z(x)} = axe^x$. It is easily seen that for $x > 0$, $z(x) > 0$, and $z(x) \uparrow \infty$ for $x \to \infty$. Now, taking natural logarithms, we obtain, for all $x > 0$, $\ln z(x) + z(x) = \ln ax + x$, or
$$\frac{z(x)}{x} = \frac{\frac{\ln ax}{x} + 1}{\frac{\ln z(x)}{z(x)} + 1}$$
which, on taking $x \to \infty$, yields the desired result since $\ln ax/x$ and $\ln z(x)/z(x)$ both go to 0.
2) Again, writing $z(x) = \mathrm{LambertW}(axe^x)$, and using the relation $\ln z(x) + z(x) = \ln ax + x$, we have
$$z(x) - x = \ln(ax) - \ln z(x) = \ln\left(\frac{ax}{z(x)}\right).$$
Since $\lim_{x\to\infty}(x/z(x)) = 1$, by Part 1 of this lemma, we obtain the desired conclusion.
3) By definition, $\mathrm{LambertW}(xe^x) = x$, and LambertW is monotone increasing for positive arguments. Hence, for $0 < a \leq 1$, $\mathrm{LambertW}(axe^x) \leq x$.
4) Follows by combining the previous two parts. ■

Proof: (Theorem 7.2) Observe that we can write $\mathrm{LambertW}(\eta(p-1)e^{\eta p})$ as $\mathrm{LambertW}((p-1/p)\eta p e^{\eta p})$, with $(p-1/p)$ being less than 1, by virtue of $p > 1$. The first two parts now follow upon using Lemma 7.1 in
$$\gamma(\eta) = \frac{\mathrm{LambertW}\left(\eta(p-1)e^{\eta p}\right) - \eta(p-1)}{\mathrm{LambertW}\left(\eta(p-1)e^{\eta p}\right)}.$$
The limit for $n\beta$ is obtained as follows. We have
$$(1-\gamma) = e^{-(n-1)\beta}.$$
Rearranging, we have
$$(n-1)\beta = -\ln(1-\gamma).$$
It follows that
$$n\beta = \frac{n}{n-1}(n-1)\beta \uparrow_{n\to\infty} \ln\left(\frac{p}{p-1}\right).$$
■

ACKNOWLEDGMENT

The authors would like to thank C. Barakat for some useful discussions.

REFERENCES

[1] F. Baccelli and S. Foss, "On the saturation rule for the stability of queues," *J. Appl. Prob.*, vol. 32, no. 2, pp. 494–507, 1995.
[2] G. Berger-Sabbatel, F. Rousseau, M. Heusse, and A. Duda, "Performance anomaly of 802.11b," presented at the IEEE INFOCOM 2003, San Francisco, CA, 2003.
[3] G. Bianchi, "Performance analysis of the IEEE 802.11 distributed coordination function," *IEEE J. Sel. Areas Commun.*, vol. 18, no. 3, pp. 535–547, Mar. 2000.
[4] F. Cali, M. Conti, and E. Gregori, "IEEE 802.11 protocol: Design and performance evaluation of an adaptive backoff mechanism," *IEEE J. Sel. Areas Commun.*, vol. 18, no. 9, pp. 1774–1780, Sep. 2000.
[5] A. Kumar, E. Altman, D. Miorandi, and M. Goyal, "New insights from a fixed point analysis of single cell IEEE 802.11 wireless LANs," presented at the IEEE INFOCOM 2005, Miami, FL, 2005.
[6] A. Kumar and D. Patil, "Stability and throughput analysis of unslotted CDMA-ALOHA with finite numer of users and code sharing," *Telecommun. Syst.*, vol. 8, pp. 257–275, 1997.
[7] B.-J. Kwak, N.-O. Song, and L. E. Miller, "Analysis of the stability and performance of exponential backoff," in *Proc. IEEE WCNC 2003*, New Orleans, LA, 2003, pp. 1754–1759.
[8] S. Mangold, S. Choi, P. May, O. Klein, G. Hiertz, and L. Stibor, "IEEE 802.11e wireless LAN for quality of service," presented at the European Wireless Conf., Florence, Italy, Feb. 2002.
[9] V. Ramaiyan, A. Kumar, and E. Altman, "Fixed point analysis of single cell IEEE 802.11e WLANs: Uniqueness, multistability and throughput differentiation," in *Proc. ACM SIGMETRICS 2005*, Banff, Alberta, Canada, Jun. 2005, pp. 109–120.
[10] A. Kumar, E. Altman, D. Miorandi, and M. Goyal, "New insights from a fixed point analysis of single cell IEEE 802.11 wireless LANs," Tech. Rep. RR-5218 INRIA, Sophia-Antipolis, France, Jun. 2004.

Anurag Kumar (F'06) received the B.Tech. degree in electrical engineering from the Indian Institute of Technology, Kanpur, and the Ph.D. degree from Cornell University, Ithaca, NY.

He was then with Bell Laboratories, Holmdel, NJ, for over six years. Since 1988, he has been with the Indian Institute of Science (IISc), Bangalore, in the Department of Electrical Communication Engineering, where he is now a Professor and the Chairman of the Department. From 1988 to 2003, he was the Coordinator at IISc of the Education and Research Network Project (ERNET), India's first wide-area packet-switching network. He is a coauthor (with D. Majunath and J. Kuri) of the advanced textbook *Communication Networking: An Analytical Approach* (Morgan-Kaufman/Elsevier, 2004). His area of research is communication networking, specifically, modeling, analysis, control and optimization problems arising in communication networks and distributed systems.

Dr. Kumar was elected a Fellow of the Indian National Science Academy (INSA) in 2006 and has been a Fellow of the Indian National Academy of Engineering (INAE) since 1998. He received the Institution of Electronics and Telecommunications Engineers (IETE) CDIL Best Paper Award (1993) and the IETE's S.V.C. Aiya Award for Telecom Education (2001). He is an Associate Editor of the IEEE TRANSACTIONS ON NETWORKING and an area editor of the IEEE Communications Surveys and Tutorials.

Eitan Altman received the B.Sc. degree in electrical engineering, the B.A. degree in physics, and the Ph.D. degree in electrical engineering from the Technion—Israel Institute of Technology, Haifa, in 1984, 1984, and 1990, respectively. In 1990, he received the B.Mus. degree in music composition from Tel-Aviv University, Tel-Aviv, Israel.

Since 1990, he has been with the National Research Institute in Informatics and Control (INRIA), Sophia-Antipolis, France. His current research interests include performance evaluation and control of telecommunication networks and, in particular, congestion control, wireless communications, and networking games.

Dr. Altman is on the editorial board of several scientific journals, including *JEDC, COMNET,* and *WINET*. He has been the General Chairman and the (Co)Chairman of the program committees of several international conferences and workshops on game theory, networking games, and mobile networks. More information can be found at http://www.inria.fr/mistral/personnel/Eitan.Altman/me.html

Daniele Miorandi received the Laurea (*summa cum laude*) and Ph.D. degrees from the University of Padova, Padova, Italy, in 2001 and 2005, respectively.

He currently holds a Researcher position at CREATE-NET, Trento, Italy. In 2003 and 2004, he spent 12 months of his doctoral thesis visiting the MAESTRO team at INRIA, Sophia-Antipolis, France. His research interests include the design and analysis of bio-inspired communication paradigms for pervasive computing environments, the analysis of TCP performance over wireless/satellite networks, scaling laws for large-scale information systems, protocols, and architectures for wireless mesh networks.

Munish Goyal received the B.S. degree in electronics and communications from the Indian Institute of Technology, Roorkee, and the M.S. and Ph.D. degrees in telecommunications from the Indian Institute of Science, Bangalore.

He was a Postdoctoral Research Fellow at the ARC Center of Excellence for Mathematics and Statistics of Complex Systems, University of Melbourne, Melbourne, Australia. He is currently with Honeywell Technologies, Bangalore. His research interests include modeling, analysis, and control problems arising in stochastic systems, especially telecommunications systems.