Collision-Minimizing CSMA and Its Applications to Wireless Sensor Networks

Y. C. Tay, Kyle Jamieson, and Hari Balakrishnan, *Member, IEEE*

***Abstract*—Recent research in sensor networks, wireless location systems, and power-saving in ad hoc networks suggests that some applications' wireless traffic be modeled as an *event-driven workload*: a workload where many nodes send traffic at the time of an event, not all reports of the event are needed by higher level protocols and applications, and events occur infrequently relative to the time needed to deliver all required event reports. We identify several applications that motivate the event-driven workload and propose a protocol that is optimal for this workload.**

Our proposed protocol, named CSMA/p^*, is nonpersistent carrier sense multiple access (CSMA) with a carefully chosen nonuniform probability distribution p^* that nodes use to randomly select contention slots. We show that CSMA/p^* is optimal in the sense that p^* is the unique probability distribution that minimizes collisions between contending stations. CSMA/p^* has knowledge of N. We conclude with an exploration of how p^* could be used to build a more practical medium access control protocol via a probability distribution with no knowledge of N that approximates p^*.

***Index Terms*—Carrier sense multiple access (CSMA), medium access control (MAC), nonpersistent, performance, poisson process, sensor networks.**

I. Introduction

CLASSICAL performance analyses of medium access control (MAC) protocols typically consider an infinitely large population of stations generating traffic with Poisson arrivals [1], [2]. More recently, however, it has been shown that in both wide-area networks (WANs) and local-area networks (LANs), traffic patterns are too bursty to be modeled as Poisson processes [3]. Furthermore, there are several examples from sensor networks and ad hoc networks that generate bursty *event-based* traffic patterns, since data is transmitted in response to external events.

1) Room Monitoring: A fire in a basement machine room of a building triggers a number of redundant temperature and smoke sensors to begin reporting the event. They all simultaneously become backlogged with the sensor reports and use a MAC protocol to arbitrate access to the medium. Higher level applications need some number of event reports that is less than the number of reporting sensors.

2) Power-Saving in Ad Hoc Networks: In a mobile ad hoc network, some *coordinator* nodes stay awake and form a routing backbone, while other nodes power down most of the time, and cannot communicate [4]. Consider the following protocol for noncoordinator neighbor discovery when one of the sleeping nodes becomes backlogged. The node wakes up unaware of its neighbors, who might have changed since it went to sleep. It sends a *poll* message. All coordinators within range of the sleeping node respond to the poll at the same time, but only one response is needed to start routing traffic from the waking source.

3) Indoor Location Systems: In the Cricket location system [5], many *beacons*, attached to walls and ceilings, broadcast their locations so that a handheld *listener* can determine its location. For fault tolerance and protection against RF or ultrasound fading, many beacons should be codeployed geographically. However, only two or three such beacons need be heard by a listener and minimizing their latency is crucial for fast update of real-time applications like navigation.

We wish to reexamine MAC design with these traffic patterns in mind. In these examples, latency, not throughput, is the performance-limiting factor. Hence, we propose the goal of minimizing the latency of the first few successful transmissions in an event-based traffic pattern that exhibits the following characteristics.

1) *An event can trigger a synchronized burst of transmissions from a large number of sensor nodes*. Note that this characteristic specifically invalidates the Poisson arrival assumption.
2) *Although a large number of nodes may decide to transmit a packet, the application at the data sink may need only a few of these packets*. Earlier work, based on packet radio, LAN, or WAN scenarios, treats all packets as equally important.
3) *In any region of space, the number of transmitting nodes can quickly change*. This follows from (1), as well as the shrinking size of sensor nodes [6] and the benefits of deploying redundant sensors.

This workload leads us to the following goal for maximizing the performance of nonpersistent carrier sense multiple access (CSMA) in sensor networks. If N sensors simultaneously and independently pick one of K slots at some point in time, what is the probability distribution on slots that yields the maximum probability of a collision-free transmission? This probability distribution, which we refer to as p^* in the rest of this paper, is optimal in the sense that it minimizes the likelihood of collisions. Furthermore, in Section III, we show that with some

Manuscript received July 15, 2003; revised February 1, 2004. This work was supported in part by the National Science Foundation under Grant 0205445. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

Y. C. Tay is with the Department of Computer Science, National University of Singapore, Kent Ridge 117543, Republic of Singapore (e-mail: tay@acm.org).

K. Jamieson and H. Balakrishnan are with the Computer Science and Artificial Intelligence Laboratory, Massachusetts Institute of Technology, Cambridge, MA 02139 USA (e-mail: jamieson@csail.mit.edu; hari@csail.mit.edu).

Digital Object Identifier 10.1109/JSAC.2004.830898

realistic assumptions, p^* does indeed minimize latency in the event-based workload.

The rest of this paper is organized as follows. Section II describes our novel probability distribution and explains how it is derived. In Section III, we examine how the optimal distribution can be used in a CSMA MAC protocol, and how it performs in such a protocol. We also derive a distribution, *Sift*, that is oblivious to N, yet approximates p^*. Section IV surveys related work and Section V concludes.

II. An Optimal Nonpersistent CSMA Distribution

We begin by describing the system model we have in mind. We assume a distributed setting where nodes have only a single radio channel available. The channel has maximum propagation delay τ. We now define the results of a nonpersistent CSMA competition over K slots.

Definition 1: We say slot r is **silent** if no node chooses that slot, and there is a **collision** if at least two nodes choose that slot. Also, a contender **wins in slot** r if and only if it is the only one to choose slot r, and all others choose later slots. Finally, there is **success** if and only if some contender wins in some slot in $1,\ldots,K$. □

In the remainder of this section, we describe the optimal distribution p^* for each node's independent choice of slot.

A. Development of the Optimal Distribution

The distribution p^* is defined in terms of a recursive function $f_s(N)$. We will see later that $f_s(N)$ is related to the probability of success when N stations compete.

Definition 2: Let s be a slot number and assume there are N contenders $N \geq 2$. Define $f_1(N)=0$ and, for $s \geq 2$

$$f_s(N)=\left(\frac{N-1}{N-f_{s-1}(N)}\right)^{N-1}.$$

□

One can prove by induction that for $s \geq 2$

$$f_{s-1}(N) < f_s(N) < 1 \tag{1}$$

i.e., $f_s(N)$ is strictly increasing with respect to s and bounded above. Moreover

$$\lim_{s\to\infty} f_s(N)=1. \tag{2}$$

Suppose there are K slots and each contender independently picks a slot r with probability p_r; we refer to the distribution $p_1,p_2,\ldots,p_K$ as p.

Definition 3: Let $\pi_p(N)$ be the probability of success when N nodes select a contention slot using probability distribution p. □

Note that the probability of success is the sum of the probabilities of success in each slot before slot K

$$\begin{aligned}\pi_p(N) &= Np_1(1-p_1)^{N-1}+Np_2(1-p_1-p_2)^{N-1}\\ &\quad+\cdots+Np_{K-1}(1-p_1-\cdots-p_{K-1})^{N-1}\\ &= N\sum_{s=1}^{K-1} p_s\left(1-\sum_{r=1}^{s}p_r\right)^{N-1}.\end{aligned} \tag{3}$$

(Since $N \geq 2$, if all sensors choose slot K, then there is a collision.) We use the following lemma to find the maximum of this probability:

Lemma 1: Given a probability distribution p, if for $j=1,\ldots,K-1$

$$\frac{\partial}{\partial p_j}\left(\frac{\pi_p(N)}{N}\right)=0$$

then

$$(N-f_i(N))p_{K-i}=(1-f_i(N))\left(1-\sum_{r=1}^{K-(i+1)}p_r\right).$$

Proof: We defer the proof of this lemma to the Appendix. □

Motivated by the above lemma, we make the following definition. This is the probability distribution that nodes should use to select CSMA contention slots in a MAC protocol.

Definition 4: The probability distribution p^* is given by

$$p_r^*=\frac{1-f_{K-r}(N)}{N-f_{K-r}(N)}(1-p_1^*-p_2^*-\cdots-p_{r-1}^*)$$

for $r=1,\ldots,K-1$.

Theorem 1: (Optimality of p^*). Suppose $N \geq 2$. Over all possible distributions p, p^* is the distribution that maximizes $\pi_p(N)$.

Proof: If $p_s=1$ for any s, then all $N(\geq 2)$ contenders collide on slot s, so the distribution cannot be optimal. If $p_s=0$ for some s, then the success probability can be increased by sharing half of $p_{s'}$ from a neighboring slot s'. Hence, the maximum must occur at an interior point ($0<p_s<1$ for all s).

Noting that p_K is determined by $p_K=1-p_1-\cdots-p_{K-1}$, $\max_{p_1,\ldots,p_{K-1}}\pi_p(N)$ must necessarily occur, where

$$\frac{\partial}{\partial p_j}\left(\frac{\pi_p(N)}{N}\right)=0 \quad \text{for} \quad j=1,\ldots,K-1. \tag{4}$$

Lemma 1 now identifies p^* in Definition 4 as the unique solution. One can further verify (by examining the second derivative of π_{p^*}) that this solution defines a maximum. □

Table I shows that p_r^* increases slowly for the initial slots (r near 1), but increases rapidly for the last few slots (r near K). For large N, the probability is concentrated in the last slot; the effect of this is to severely limit the number of contenders for the first $K-1$ slots and, thus, improve the probability of a success. For example, if $K=32$ and $N=1024$, only six contenders are expected to choose from the first 31 slots, so it is not surprising that the probability of success is 0.94.

It is easy to see that $p_r^*=(1/K)$ for all r if and only if $N=2$, so the uniform distribution is optimum exactly when there are two contenders. Thus, although many protocols' bounded exponential backoff mechanism (see Section I) adjusts the contention window size to suit the number of contenders, the uniform distribution they use remains suboptimal.

B. Properties of $f_s(N)$ and p^*

We next provide some intuition for $f_s(N)$, and show that the maximum value for $\pi_p(N)$ is $f_K(N)$.

TABLE I

EXAMPLES OF THE OPTIMAL DISTRIBUTION. (TOP) OPTIMAL DISTRIBUTIONS FOR $K = 8$. NOTE THE FLAT TAIL ON THE LEFT AND THE STEEP INCREASE ON THE RIGHT. (BOTTOM) $K = 32$. NOTE THE HIGH SUCCESS PROBABILITY DESPITE THE LARGE CHANGE IN N (SEE ALSO FIG. 1)

	p_1^*	p_2^*	p_3^*	p_4^*	p_5^*	p_6^*	p_7^*	p_8^*	
$N = 16$	0.015	0.017	0.019	0.022	0.027	0.036	0.054	0.810	$\pi_{p^*}(16) = 0.80$
$N = 128$	0.0018	0.0021	0.0024	0.0029	0.0036	0.0049	0.0077	0.9746	$\pi_{p^*}(128) = 0.79$

	p_1^*	p_2^*	p_3^*	$\cdots$	p_{29}^*	p_{30}^*	p_{31}^*	p_{32}^*	
$N = 64$	0.00095	0.00098	0.00101	$\cdots$	0.00691	0.00926	0.01448	0.91222	$\pi_{p^*}(64) = 0.942$
$N = 1024$	0.000059	0.000061	0.000063	$\cdots$	0.000456	0.000615	0.000972	0.994297	$\pi_{p^*}(1024) = 0.941$

Definition 5: If p is a slot probability distribution, let p'_r be the probability of choosing slot r conditioned on not choosing any slot before r.

Corollary 1: For the distribution p^*:

i) $p_r^{*\prime} = (1 - f_{K-r}(N))/(N - f_{K-r}(N))$ for $r = 1, \ldots, K-1$.

ii) $f_{K-r}(N) = (1 - p_{r+1}^{*\prime})^{N-1}$ for $r = 0, \ldots, K-2$.

iii) $f_{K-r}(N) = \Pr(\text{there is a winner} \mid \text{no contender chooses any slot before } r+1)$ for $r = 0, \ldots, K-1$.

iv) The maximum value of $\pi_p(N)$ is $\pi_{p^*}(N) = f_K(N)$.

Proof: We prove each of the four propositions individually.

i) This follows from $p_r^{*\prime} = (p_r^*)/(1 - p_1^* - \cdots - p_{r-1}^*)$ and Definition 4.

ii) By the definition of $f_s(N)$ and (i), $f_{K-r}(N) = ((N-1)/(N - f_{K-r-1}(N)))^{N-1} = (1 - p_{r+1}^{*\prime})^{N-1}$.

iii) By induction on $r = K-1, K-2, \ldots, 1, 0$. For $r = K-1$, if no contender chooses any slot before $r+1 = K$, then all contenders collide $(N \geq 2)$ in slot K, so the probability of a winner is $0 = f_1(N)$.

Assume iii) is true for $r = \ell$. For $r = \ell - 1$, if no contender chooses any slot before ℓ, then there is a winner if and only if the winner chooses slot ℓ, or no contender chooses slot ℓ and the winner picks a later slot. Hence, by the hypothesis, the probability that there is a winner given that no one chooses any slot before ℓ is $Np_\ell^{*\prime}(1 - p_\ell^{*\prime})^{N-1} + (1 - p_\ell^{*\prime})^N f_{K-\ell}(N) = (1 - p_\ell^{*\prime})^{N-1} = f_{K-(\ell-1)}(N)$. This completes the induction.

iv) $\pi_{p^*}(N) = \Pr(\text{there is a winner}) = f_K(N)$ by iii).

□

Fig. 1 shows that, as N increases, $\pi_{p^*}(N)$ drops initially, then becomes almost constant. We see that, for any fixed K, it is possible to maintain the probability of success as N increases by suitably adjusting the distribution.

Let $\ell_{p^*}(N)$ be the expected successful slot number when the slot is chosen with p^*. Since p^* is uniform for $N = 2$, we have

$$\ell_{p^*}(2) = \sum_{j=1}^{K} j \frac{2}{K}\left(1 - \frac{j}{K}\right) = \frac{K^2 - 1}{3K}. \tag{5}$$

It turns out that $\ell_{p^*}(N) \approx \ell_{p^*}(2)$ for any N and K, as illustrated in Table II.

C. A K-Stage Interpretation of p^*

For $K = 2$, the optimal distribution is

$$p_1^* = \frac{1}{N} \quad \text{and} \quad p_2^* = \frac{N-1}{N}. \tag{6}$$

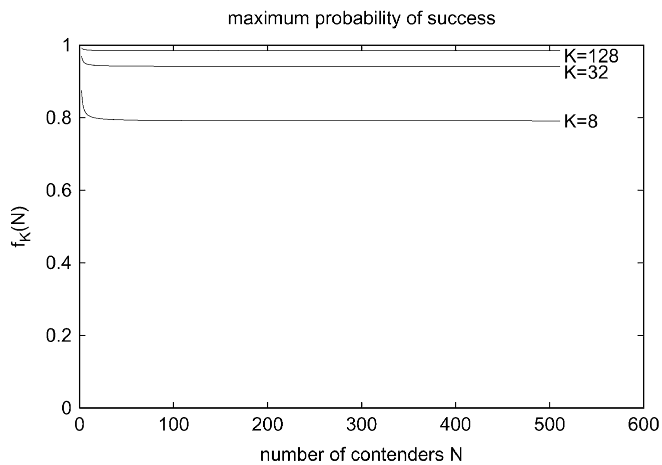

Fig. 1. After an initial drop, the maximum probability of success $\pi_{p^*}(N)$ becomes flat as N increases.

TABLE II

EXPECTED SLOT NUMBER FOR SUCCESSFUL TRANSMISSION, USING THE OPTIMAL DISTRIBUTION WHEN $N = 2, 8, 1024$. NOTE THAT, FOR A FIXED K, $\ell_{p^*}(N)$ IS ALMOST CONSTANT WITH RESPECT TO N

K	2	16	32	64	128
$\ell_{p^*}(2)$	0.5	5.3	10.7	21.3	42.7
$\ell_{p^*}(8)$	0.4	5.2	10.6	21.4	42.7
$\ell_{p^*}(1024)$	0.4	5.2	10.6	21.3	42.8

The intuition here is that if $Np_1^* > 1$, then more than one contender is expected to choose the first slot and so collide there; if $Np_1^* < 1$, then less than one contender is expected to choose the first slot, so we expect almost all N contenders to collide in the second slot. Thus, p_1^* is optimal when $Np_1^* = 1$.

For $K > 2$, the intuition for p_1^* is less clear. One can view it as the result of the following optimization. By the inductive step in Corollary 1 (iii)

$$f_K(N) = Np_1^*(1 - p_1^*)^{N-1} + (1 - p_1^*)^N f_{K-1}(N). \tag{7}$$

By Corollary 1 (iv), $f_K(N)$ and $f_{K-1}(N)$ are both maximum, so $(\partial f_K(N))/(\partial p_1^*) = 0 = (\partial f_{K-1}(N))/(\partial p_1^*)$. Differentiating (7), we get $0 = N(1 - p_1^*)^{N-2}((1 - p_1^*) - (N-1)p_1^* - (1 - p_1^*)f_{K-1}(N))$. One solution to this equation is $p_1^* = 1$, which is clearly not optimum; the other solution says that the optimal probability for choosing the first of K slots is

$$\frac{1 - f_{K-1}(N)}{N - f_{K-1}(N)}. \tag{8}$$

The choice of p_1 affects the probability of succeeding in the later $K-1$ slots, so $f_{K-1}(N)$ here represents the feedback effect from those slots. For $K = 2$ (and $N \geq 2$), success in slot 2 is not possible; hence, there is no feedback, $f_1(N) = 0$ as in Definition 2 and we get $p_1^* = (1/N)$, as noted previously (6).

We can now understand the optimal distribution (Definition 4) for slot selection as a K-stage process: In stage 1, each contender picks slot 1 with probability $p_1^{*\prime} = p_1^* = (1 - f_{K-1}(N))/(N - f_{K-1}(N))$, according to the above observation (8). If no one chooses slot 1, then each contender picks slot 2 with the optimal first-slot probability for $K - 1$ slots; by the same observation (8), this probability is $(1 - f_{K-2}(N))/(N - f_{K-2}(N))$ and, thus, $p_2^{*\prime} = (1 - f_{K-2}(N))/(N - f_{K-2}(N))$. Repeating this decision procedure yields Corollary 1 (i) and, equivalently, the optimal distribution in Definition 4.

Note that since (7) is derived from Corollary 1, this is not an alternative proof of the theorem. Rather, we are merely reinterpreting the distribution for a one-time slot selection as a K-stage decision.

III. Applications of the Optimal Distribution

We now use the optimal distribution to compare the fundamental performance limits of both persistent and nonpersistent CSMA, with respect to the latency of the first transmission, when N stations become simultaneously backlogged in a previously quiescent network.

Consider first an arbitrary nonpersistent CSMA protocol that picks one of K slots for transmission. Let T_{slot} and T_{packet} be the time duration for a slot and a packet transmission (including any necessary interframe spacing), respectively. Define the latency $L_{\text{CSMA}}(N)$ to be the expected delay for a successful transmission when there are N contenders.

If there is a collision, then the delay is at least T_{packet}, so

$$L_{\text{CSMA}}(N) \geq (1 - \pi_p(N))T_{\text{packet}} \tag{9}$$

where p is the slot selection distribution used by the protocol. Since $\pi_p(N) \leq f_K(N)$, we get

$$L_{\text{CSMA}}(N) \geq (1 - f_K(N))T_{\text{packet}}. \tag{10}$$

This is a general lower bound, so it is weak (see Fig. 2).

One way to strengthen the lower bound (10) is to specify the distribution p. For the popular uniform distribution, $\pi_p(N)$ is given by $N(1/K)(1-(1/K))^{N-1}+N(1/K)(1-(2/K))^{N-1} + \cdots + N(1/K)(1 - (K-1/K))^{N-1} < (N/K)(e^{-(N/K)} + \cdots + e^{-(K-1)(N/K)}) < (N/K)1/(e^{(N/K)} - 1)$. Inequality (9) now gives a lower bound on latency for the uniform distribution:

$$L_{\text{uniform}}(N) \geq \left(1 - \frac{N}{K(e^{\frac{N}{K}} - 1)}\right) T_{\text{packet}}. \tag{11}$$

This lower bound is much stronger than (10), as illustrated in Fig. 2.

To see how close we can get to the lower bound (10), we need to specify what the protocol does if there is a failure. Fig. 3 defines CSMA/p, a nonpersistent CSMA using p as the distribution with which nodes pick contention slots. In CSMA/p, nodes choose new contention slots whenever they become backlogged, or if they carrier sense the medium clear after previous activity.

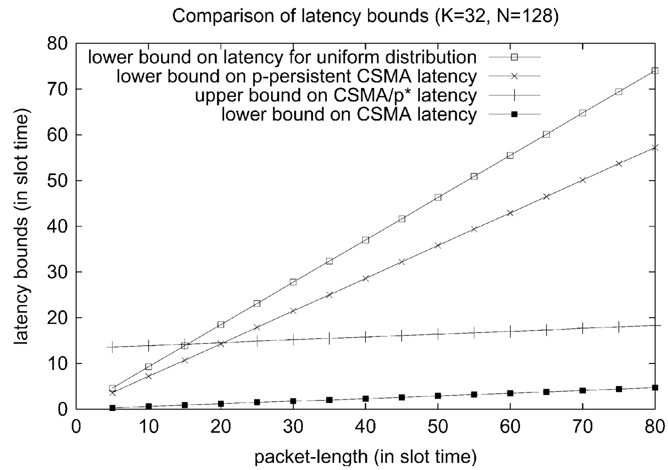

Fig. 2. This graph compares the lower bounds $L_{\text{uniform}}(N)$ in (11), $L_{\text{persistent}}(N)$ in (13), $L_{\text{CSMA}}(N)$ in (10), and the upper bound $L_{\text{CSMA}/p^*}(N)$ in (12). Note that the latencies for the uniform distribution and p-persistent protocols may be higher than their respective lower bounds.

repeat
    if carrier sense clear **and** backlogged **do**
        $r \in_p [1, K]$
        wait r contention slots
        if carrier sense clear **then**
            transmit
        end if
    end if
end repeat

Fig. 3. CSMA/p: Nonpersistent CSMA using p. The notation $x \in_p [a, b]$ signifies choosing x at random using distribution p over the interval $[a, b]$.

Consider now CSMA/p^*. If there is a successful transmission in one round of contention, its latency $L_{\text{CSMA}/p^*}(N)$ is $\ell_{p^*}(N)T_{\text{slot}}/\pi_{p^*}(N)$. If there is a collision, $L_{\text{CSMA}/p^*}(N)$ is at most $KT_{\text{slot}} + T_{\text{packet}} + L_{\text{CSMA}/p^*}(N)$. Hence, $L_{\text{CSMA}/p^*}(N) < \pi_{p^*}(N)(\ell_{p^*}(N))/(\pi_{p^*}(N))T_{\text{slot}} +(1 - \pi_{p^*}(N))(KT_{\text{slot}} + T_{\text{packet}} + L_{\text{CSMA}/p^*}(N))$; using $\ell_{p^*}(N) \approx \ell_{p^*}(2)$ and (5), we get

$$L_{\text{CSMA}/p^*}(N) < \frac{K^2 - 1}{3Kf_K(N)}T_{\text{slot}} + \left(\frac{1}{f_K(N)} - 1\right)(KT_{\text{slot}} + T_{\text{packet}}). \tag{12}$$

This upper bound is illustrated in Fig. 2.

The K-stage interpretation (described in Section II-C) is similar to a p-persistent slotted CSMA [2] except that instead of transmitting with the same probability in each slot indefinitely, each contender changes its probability of transmission after each silent slot, and will transmit with certainty after K slots.

As in the optimal distribution for $K = 2$ (6), the probability of a successful transmission for this p-persistent protocol is maximum when $p = 1/N$. Now, let $L_{\text{persistent}}(N)$ be the latency for this protocol, using $p = 1/N$. If there is a successful transmission in the first slot, the latency is 0. If all contenders skip the slot, the latency is $T_{\text{slot}} + L_{\text{persistent}}(N)$. If there is a collision, the latency is $T_{\text{packet}} + L_{\text{persistent}}(N)$. Thus,

$L_{\text{persistent}}(N) = (1-(1-(1/N))^{N-1})L_{\text{persistent}}(N) + (1-(1/N))^N T_{\text{slot}} + (1-(1-(1/N))^{N-1} - (1-(1/N))^N)T_{\text{packet}}$, so

$$\begin{aligned} L_{\text{persistent}}(N) &> \frac{1-\left(1-\frac{1}{N}\right)^{N-1} - \left(1-\frac{1}{N}\right)^{N}}{\left(1-\frac{1}{N}\right)^{N-1}} T_{\text{packet}} \\ &\approx (e-2)T_{\text{packet}}. \end{aligned} \tag{13}$$

This lower bound is shown in Fig. 2.

Fig. 2 shows that the upper bound for L_{CSMA/p^*} grows like the general lower bound for L_{CSMA}, and is lower than the lower bounds L_{uniform} and $L_{\text{persistent}}$ except for very small packet sizes. In other words, for sufficiently large packet sizes, CSMA/p^* has a smaller latency than any p-persistent protocol and any protocol using the uniform distribution.

Indeed, the same is true for an arbitrary distribution p that is different from p^*, if we fix the protocol to be CSMA/p. To see this, let $\ell_p(N)$ be the expected successful slot number and $\ell'_p(N)$ the expected slot number where the collision begins if there is a failure. Then, $L_{\text{CSMA}/p}(N) = \ell_p(N)T_{\text{slot}} + (1-\pi_p(N))(\ell'_p(N)T_{\text{slot}} + T_{\text{packet}} + L_{\text{CSMA}/p}(N))$, so $L_{\text{CSMA}/p}(N) = (\ell_p(N))/(\pi_p(N))T_{\text{slot}} + ((1)/(\pi_p(N)) - 1)\ell'_p(N)T_{\text{slot}} + +((1)/(\pi_p(N)) - 1)T_{\text{packet}}$ and

$$\begin{aligned} L_{\text{CSMA}/p}(N) &- L_{\text{CSMA}/p^*}(N) \\ &= t_p(N)T_{\text{slot}} + \left(\frac{1}{\pi_p(N)} - \frac{1}{f_K(N)}\right)T_{\text{packet}} \end{aligned} \tag{14}$$

where $t_p(N) = (\ell_p(N))/(\pi_p(N)) + ((1/\pi_p(N)) - (1)\ell'_p(N) - (\ell_{p^*}(N))/(f_K(N)) - ((1)/(f_K(N)) - 1)\ell'_{p^*}(N)$.

Now, $(1)/(\pi_p(N)) - (1)/(f_K(N))$ is positive because p^* is the unique distribution that maximizes $\pi_p(N)$, so for sufficiently large T_{packet}, (14) implies $L_{\text{CSMA}/p}(N) - L_{\text{CSMA}/p^*}(N) > 0$, i.e., CSMA/$p^*$ has smaller latency than any CSMA/p protocol.

Both the nonpersistent and persistent CSMA proposals examined above require knowledge of N. They, thus, give up one of the main advantages of CSMA. In the following sections, we consider ways to ameliorate this unfortunate fact.

A. Using the Optimal Distribution in a MAC Protocol

Suppose n contenders use p^* for slot selection, but with an N value that is possibly different from n. Then, the success probability is $f_K(N)$ exactly when $n = N$; for all other n, the success probability is smaller. This is illustrated in Fig. 4.

The graph shows that, for large N, the success probability drops sharply to the left, but gently to the right. In other words, the penalty is severe for overestimating n (i.e., $N > n$), but minor for underestimating n (i.e., $N < n$).

When N overestimates n, p^*_K is too large (see Table I), and we are likely to get silence in the first $K-1$ slots. This is illustrated in Fig. 5, which shows that most failures occur with silence in the first $K-1$ slots if $n \ll N$.

When N underestimates n, there are more contenders than is provided for by p^*, so a collision is likely to occur in the first $K-1$ slots. If the stations can infer collisions, (by lack of the receipt of an acknowledgment, for example) then the additive-increase/multiplicative-decrease (AIMD) scheme [7] of Fig. 6 suggests itself.

In effect, this AIMD scheme is adjusting the slot selection distribution to suit the level of contention. However, since events trigger synchronous bursts of contention, this adjustment may be slow and, worse still, waste energy through colliding transmissions and code execution. In this context, it is better to use a distribution that does not require an estimate of the number of sensors. We present such a distribution next.

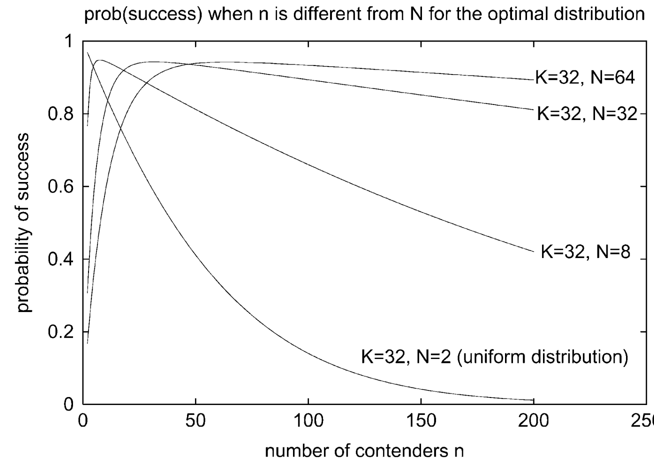

Fig. 4. When the number of contenders n is different from the value of N used by the optimal distribution, the probability of success is maximum at $n = N$. On the left-hand side of the maximum $(n < N)$, N is an overestimate of n; on the right-hand side $(N < n)$, N is an underestimate of n. Note that the uniform distribution $p_r = (1/K)$ is optimum only when $N = 2$ (as shown in Section II.A).

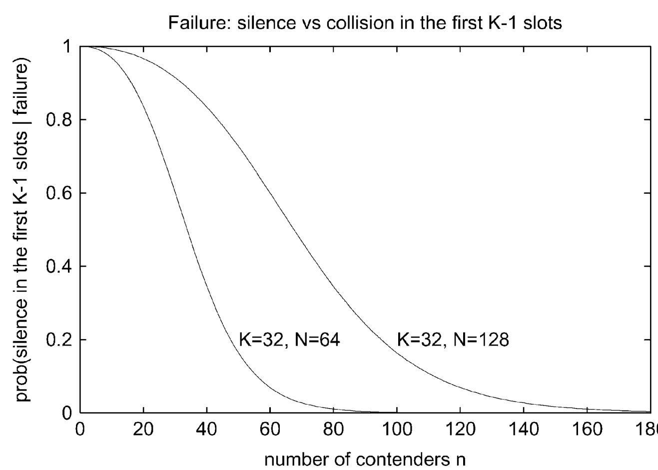

Fig. 5. Failure occurs if the first $K-1$ slots are silent—i.e., all $N(\geq 2)$ contenders choose the last slot—or they contain a collision. This graph shows that, when $n \ll N$, most of the failures are from silence; for $n > N/2$, failures are dominated by collisions in the first $K-1$ slots.

```
if failure then
    if silence in the first K − 1 slots then
        N ← N/2
    else
        N ← N + 1
    end if
end if
```

Fig. 6. An AIMD policy based on testing whether there was silence in the first $K-1$ slots.

B. *Sift: A Distribution to Approximate* CSMA/p^*

To function in a wireless network, CSMA/p^* requires knowledge of, or at least an estimate of, the number of contenders. For practicality, we seek a protocol that does not require such an estimate but which, nonetheless, has a success probability that is close to optimal.

Since we seek to approximate $f_K(N)$, the natural starting point is p^*. From (23), in the Appendix, we get

$$(N-1)p^*_{(K-\ell)-1} = (1 - f_{\ell+1}(N)) \left(1 - \sum_{j=1}^{(K-\ell)-1} p^*_j\right) \quad (15)$$

or setting $r = K - \ell$

$$(N-1)p^*_{r-1} = (1 - f_{K-r+1}(N)) \left(1 - \sum_{j=1}^{r-1} p^*_j\right) \quad (16)$$

This and Definition 4 give

$$\begin{aligned}\frac{p^*_{r-1}}{p^*_r} &= \frac{1 - f_{K-r+1}(N)}{1 - f_{k-r}(N)} \frac{N - f_{K-r}(N)}{N-1}\\ &\approx \frac{1 - f_{K-r+1}(N)}{1 - f_{k-r}(N)}, \quad \text{for large } N\\ &= \frac{1 - \left(1 - \frac{1-f_{K-r}(N)}{N - f_{K-r}(N)}\right)^{N-1}}{1 - f_{K-r}(N)}, \quad \text{by Definition 2}\\ &\approx \frac{1 - e^{-(1-f_{K-r}(N))}}{1 - f_{K-r}(N)}, \quad \text{for large } N\\ &\approx \frac{1}{2}(1 + f_{K-r}(N))\end{aligned}$$

since $1 - f_{K-r}(N) \approx 0$ for large $K - r$.

By the monotonicity [Inequality (1)] and limit (2) properties of $f_s, f_s(N)$ changes slowly for large s. Hence, for large $K, f_{K-r+1}(N) \approx f_{K-r}(N)$ if r is small. It follows from the above derivation that $(p^*_{r-1})/(p^*_r)$ is approximately constant for the initial slots (small r), i.e., the left-hand tail of the optimal distribution is approximately geometric. This leads us to the truncated geometric distribution

$$g_r = \frac{(1-\alpha)\alpha^K}{1-\alpha^K} \cdot \alpha^{-r}, \quad \text{for } r = 1, \ldots, K \quad (17)$$

where $\alpha = g_{r-1}/g_r$ is a distribution parameter, $0 < \alpha < 1$. This distribution has g_r increasing exponentially with r, like in Table I.

We have used this distribution to design Sift, a MAC protocol for wireless sensors [8]. Essentially, the protocol is CSMA/g. This gives a very simple protocol, with no adjustment of window size K to suit N, nor suspension of timers, like in 802.11. For sensors, simplicity is essential, energy-wise. (Note that, for Sift, the probability of success in slot r decreases rapidly with r, so timer suspension is of marginal benefit and can even hurt, since collisions waste energy and add to latency.)

Using an argument similar to the K-stage interpretation (Section II-C), the parameter α is determined by $\alpha = M^{-(1)/(K-1)}$, where M is the maximum number of contenders (here, M is the analog of 802.11's maximum window size); one can show that, thus configured, the success probability will be approximately $f_K(N)$ for any $N \leq M$.

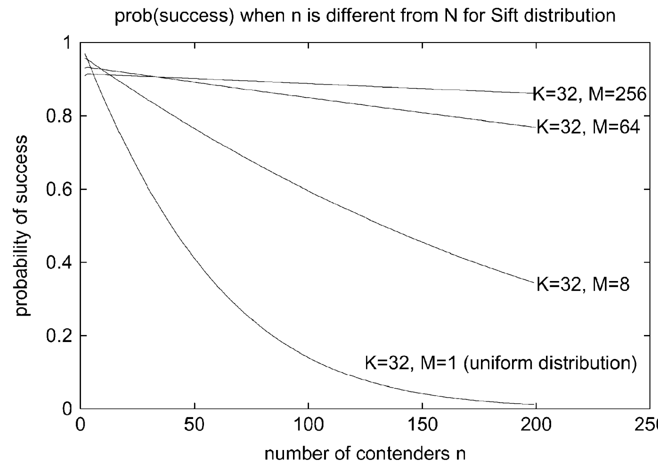

Fig. 7. For this graph, the Sift distribution is configured with $\alpha = M^{-(1)/(K-1)}$. (Note that, in (17), $\lim_{\alpha \to 1} g_r = (1/K)$.) This graph shows the success probability when the number of contenders is different from M, so it can be compared with Fig. 4 for the optimal distribution. For $M = 1$, we get the uniform distribution, so the curve is the same as the $N = 2$ curve in Fig. 4. When $n < M$, the success probability is high, as intended. If $n > M$ (i.e., n exceeds the maximum that is used for the configuration), the success probability nonetheless degrades gracefully, without any sudden drop.

This claim is supported by Fig. 7, which plots the results of an experiment in which N sensors choose slots using the distribution in (17), with various values of α. The graph shows that, without using N, the success probability for Sift remains high for all $N \leq M$. Intuitively, when N is near M, the success comes from the earlier slots (r near 1), where the probabilities are tuned for large N. For small N, however, the probabilities in the early slots are too small, so they are unused and the success occurs in one of the later slots (r near K).

Note that in each instance, although we engineered the Sift distribution for some maximum number of sensors M, its performance degrades gracefully when the true number of contending stations exceeds M.

C. *Scalability of the Sift Distribution*

The Sift distribution (17) has the desirable property that, if $\alpha = M^{-(1)/(K-1)}$, then for any $1 \leq N \leq M$, the probability of success $\pi_g(N)$ is near the optimum. Since one motivation for this distribution is that M can be large for sensor networks, we should check the Sift scheme for scalability.

We formulate the scalability question this way: If M increases to M'', what should K be so that $\pi_g(N)$ is near optimum for any $1 \leq N \leq M''$? Ideally, a big increase in M to M'' should only require a small increase in K.

Let K'' be the new K, and $\alpha'' = M''^{-(1/K'')-1}$. We now show that we can answer the above question by choosing K'' so that $\alpha = \alpha''$. In particular, this means that an exponential increase in N'' only requires a linear increase in K.

First, let p_r'' be the probability (17) when K is K'' and $\alpha = \alpha''$. For convenience, assume M divides M'', and first consider N'' such that $N'' = (N/M)M''$ for some $2 \leq N \leq M$. Then

$$\begin{aligned} N''p_r'' &= N'' \frac{(1-\alpha)\alpha^{K''}}{1-\alpha^{K''}}\alpha^{-r}, \quad \text{since } \alpha'' = \alpha \\ &\approx N''\alpha^{K''-1}\frac{1-\alpha}{1-\alpha^K}\alpha^{1-r}, \quad \text{for small } \alpha^K \quad \text{and} \quad \alpha^{K''} \\ &= \frac{N''}{M''}\frac{1-\alpha}{1-\alpha^K}\alpha^{1-r}, \quad \text{since} \quad \alpha = \alpha'' = M''^{-\frac{1}{K''-1}} \\ &= \frac{N}{M}\frac{1-\alpha}{1-\alpha^K}\alpha^{1-r}, \quad \text{since } N'' = \frac{N}{M}M'' \\ &= N\alpha^{K-1}\frac{1-\alpha}{1-\alpha^K}\alpha^{1-r}, \quad \text{since } \alpha = M^{-\frac{1}{K-1}} \\ &= Np_r. \end{aligned} \tag{18}$$

For the larger configuration (with N'' and K'') and $j < K$,

$$\begin{aligned} &\Pr(\text{success in slot } j \text{ given } N'' \text{ contenders}) \\ &\quad = N''p_j''(1-(p_1''+\cdots+p_j''))^{N''-1} \\ &\quad \approx N''p_j''e^{-N''(p_1''+\cdots+p_j'')} \quad \text{for large } N'' \\ &\quad \approx Np_je^{-N(p_1+\cdots+p_j)} \quad \text{by (18)} \\ &\quad \approx Np_j(1-(p_1+\cdots+p_j))^{N-1} \\ &\quad = \Pr(\text{success in slot } j \text{ given } N \text{ contenders}) \end{aligned}$$

in the smaller configuration (with N and K). It follows that $\pi_g(N'')$ (the probability of success in any slot) in the larger configuration is approximately equal to $\pi_g(N)$ in the smaller configuration, if $N'' = (N/M)M''$.

Now, $\pi_g(N)$ is near the optimum for $N = 2, 3, \ldots, M$, so $\pi_g(N'')$ is near the optimum for $N'' = (2/M)M''$, $(3/M)M'', \ldots, (M/M)M''$. (Note: The optimum values for N and for N'' are approximately equal—see Fig. 1) Since $\pi_g(N'')$ does not vary drastically between $N'' = (N/M)M''$ and $N'' = (N+1/M)M''$, we conclude that $\pi_g(N'')$ is near the optimum for any $2 \leq N'' \leq M''$.

Finally, for $\alpha'' = \alpha$, we require $M''^{-(1)/(K'')-1} = M^{-(1)/(K-1)}$, i.e., $K'' = (K-1)\log_M M'' + 1$. Thus, if $M'' = M^x$, then $K'' = (K-1)x + 1$; i.e., to maintain near-optimality for an unknown (and variable) number of contenders, an exponential increase in M only requires a linear increase in K.

For example, suppose we have $K = 32$ and configured Sift for $M = 128$; to reconfigure for a maximum number of contenders $M'' = 16\,384 = 128^2$, we require just $K'' = 63$. Fig. 8 shows that the resulting success probability is still close to optimum, despite the enormous range in number of contenders and the limited number of slots.

We have shown a distribution that, without changing the number of slots K, supports values of N that are large in relation to K. Furthermore, a MAC protocol using this distribution does not require knowledge of N, and so is robust with respect to quick changes in N. Finally, this distribution has a success rate close to optimal.

D. Energy Consumption

A primary consumer of energy in a wireless network is time spent listening for, but not receiving packets (idle energy).

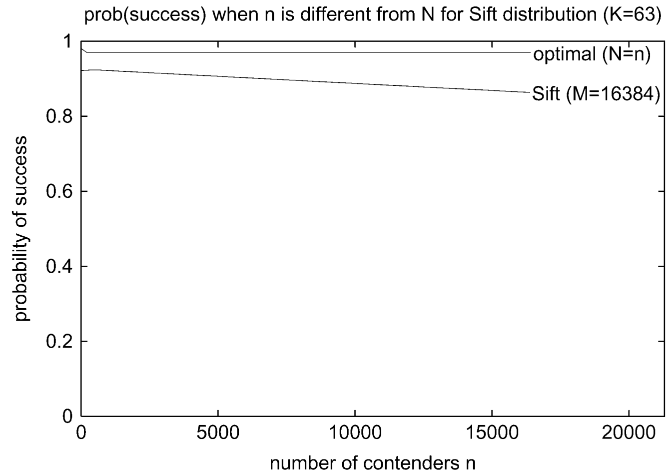

Fig. 8. Sift is scalable: An exponential increase in M only requires a linear increase in K. The success probability remains close to optimum despite the enormous range in number of contenders and the limited number of slots.

Span[4], GAF [9], and S-MAC [10] address this issue, and are compatible with CSMA/p^* or Sift simply by modulating the uniform backoff distribution of those protocols. Packets received by an overhearing node are another energy drain [10], [11]. Again, no changes to the above power-saving protocols are required to take advantage of our novel backoff distributions. Finally, collisions have been shown to be costly in terms of energy [11], and so by minimizing or reducing the number of collisions, our distributions reduce the energy consumption of any CSMA-based MAC even further.

E. Hidden Terminals and CSMA Slot Time

In this paper, we consider slotted CSMA, where the slot time T_{slot} is equal to the sum of τ, the time required to sense the channel idle, the time required to switch the radio from receive to transmit mode, and any other radio processing delay. Thus, if two stations can carrier sense each others' transmissions, then they collide if and only if they pick the same contention slots. However, this statement is not true if the stations cannot carrier sense each others' transmissions.

Modern spread-spectrum radios have a carrier-sensing range approximately twice that of their transmission range [12], [13], making it more likely that a node will carrier-sense a transmission that can interfere at the receiver of its transmission. This lessens the frequency of hidden terminals. For large packets, the RTS/CTS exchange mitigates collisions between hidden terminals. In the case of collisions between small data packets among hidden terminals, receivers can arbitrate between hidden terminals as CODA [14] proposes, or hidden terminals can vary their transmit phases with respect to one other to avoid collisions [15].

In MACAW [16], Bharghavan proposed that the slot time be equal to the duration of an RTS packet transmission time plus T_{slot}. The 802.11 standard [12] sets the slot time equal to T_{slot}, but increases the contention window size using bounded exponential backoff. These designs offer resilience against hidden terminals at the expense of performance [8].

IV. Related Work

Gao and Rubin study slotted Aloha [1] with an infinite population of users generating traffic according to a multiplicative multifractal process, parameterized for different levels of burstiness [17]. They find a performance degradation when traffic is bursty, supporting our argument for a MAC layer that is designed for bursty traffic.

A. CSMA-Based Protocols

CSMA combined with stabilization techniques such as binary exponential backoff [18] has been proposed for the wired Ethernet [18], 802.11 [12], and MACAW [16] wireless LANs, and for sensor networks [10]. Such protocols have had enormous practical success in wireless networks, but as we show [8], do not scale well under the event-based workload.

Calì *et al.* analyze CSMA from the standpoint of throughput. They propose replacing the uniform-distribution contention window of 802.11 with a p-persistent backoff protocol [19]. By estimating the population size, they choose p to maximize system throughput when *all* nodes *always* have a packet ready for transmission. They show that 802.11 yields suboptimal throughput under this workload, and that their algorithm can approach optimal throughput under the same conditions.

Cai *et al.* [20] propose a polynomial distribution for nonpersistent CSMA contention slot selection. Their distribution is optimal over the space of all polynomial functions, whereas our p^* is optimal over the space of all probability density functions. They maximize the parameter of the polynomial distribution numerically, whereas we provide closed form expressions for p^* and for α. Finally, they use Poisson arrivals and throughput measurements to evaluate their protocol, whereas we propose the event-based workload, and evaluate our protocol using this workload.

The HIPERLAN standard [21] for wireless LANs uses a truncated geometric probability distribution in the "elimination" phase of its contention protocol. Cho *et al.* [22] describe and analyze HIPERLAN's MAC protocol in detail. In this paper, we propose the use of traditional CSMA, where immediately following a busy channel, the first station to break the silence wins access to the medium. In contrast, HIPERLAN stations transmit noise bursts of varying length after the medium becomes idle, and the station that ceases its noise burst *last* wins access to the medium. Our approach compares favorably with HIPERLAN for two reasons. HIPERLAN's noise bursts raise the overall noise floor of the network when there are many stations, and consume more power than listening for the same amount of time on most radio hardware.

Ethernet [18] uses CSMA with carrier detection (CSMA/CD) to detect collisions, and bounded exponential backoff (BEB) to resolve them when N is large. This approach does not scale under our traffic model, since as N increases, many rounds of exponential backoff are required until even the first successful transmission. MACAW [16] and FAMA [23] also use the same BEB algorithm, except that in wireless networks, there is no collision detection. Still, the same number of rounds are required to resolve a collision, hence, MACAW also does not scale under our traffic model. Finally, like Ethernet and MACAW, IEEE 802.11 [12] also uses a uniform contention window with BEB controlling its size. Tree-splitting collision resolution [1] resolves collisions after they occur. FAMA-CR [24] is an example of a CSMA protocol that uses tree-splitting contention resolution.

Tree-splitting schemes require nodes participating in the contention-resolution protocol to assume that transmissions sent to other nodes in the same phase of contention resolution were successfully received. Since interference is a property of the receiver, this assumption is not always true, especially in a noisy channel. Tree-splitting schemes also incur the performance penalty of a probable collision the first time many nodes become backlogged. This results in longer latencies under our workload.

B. TDMA-Based Protocols

For a bursty workload, simple round-robin time-division multiple access (TDMA) [25] is highly suboptimal, resulting in many wasted data slots. Furthermore, there have been many independent proposals for conserving power in CSMA sensor networks [10], [26], and in multihop ad hoc networks in general [4], [9].

Probabilistic time division (PTD) [27] is a TDMA-like scheme in which stations transmit in each TDMA slot with a given probability. Each station chooses one TDMA slot in each round with a fixed probability a. By tuning a, the authors achieve a compromise between TDMA and pure random access. Our proposal differs from PTD because we compute an optimal probability distribution on *contention slots*, which in practical networks are several orders of magnitude smaller than TDMA data slots. Our work is purely contention-based, and unlike PTD, does not require slot-synchronization among stations.

Time-spread multiple-access (TSMA) protocols assign each node a unique code that deterministically specifies the time slots in which the node has the right to transmit. GRAND [28] is a TSMA algorithm where performance depends on the maximum node degree of the network $D_{\max}$. In a highly variable or completely connected network, $D_{\max}$ could be $O(N)$, where N is the number of nodes in the network. This would result in a $O(N^2)$ frame length, which wastes large numbers of TDMA slots. T-TSMA [29] runs a number TSMA protocols simultaneously, using round-robin TDMA to switch between protocols. Each protocol is tuned for nodes with neighborhood sizes equal to increasing powers of two. While T-TSMA scales in terms of asymptotics, any threaded protocol wastes slots running many protocols at once.

V. Conclusion

Instead of a workload consisting in a Poisson arrival process, we have studied MAC latency when N nodes become simultaneously backlogged at some point in time. This workload is important in the context of an event-driven sensor network, when N sensors simultaneously sense an event of interest from the outside world. Under this workload, we have derived the optimal backoff distribution p^* for a nonpersistent CSMA protocol, in which every node chooses a contention slot according to p^*.

We have compared optimal nonpersistent CSMA with persistent CSMA, discussed possible implementation directions for the optimal protocol, and finally discussed Sift. Sift's backoff distribution approximates p^* no matter how many stations are backlogged, obviating the need to track this figure with a contention window and bounded exponential backoff.

APPENDIX

We now restate and prove Lemma 1.

Lemma 1: Given a probability distribution p, if for $j = 1, \ldots, K-1$

$$\frac{\partial}{\partial p_j}\left(\frac{\pi_p(N)}{N}\right) = 0$$

then

$$(N - f_i(N))p_{K-i} = (1 - f_i(N))\left(1 - \sum_{r=1}^{K-(i+1)} p_r\right). \quad (19)$$

Proof: Recall from (3) that

$$\pi(N) = N \sum_{s=1}^{K-1} p_s \left(1 - \sum_{r=1}^{s} p_r\right)^{N-1}.$$

Now

$$\frac{\partial p_s}{\partial p_j} = \begin{cases} 0, & \text{if } s \neq j \\ 1, & \text{if } s = j \end{cases}$$

and

$$\frac{\partial}{\partial p_j} \sum_{r=1}^{s} p_r = \begin{cases} 0, & \text{if } s < j \\ 1, & \text{if } s \geq j \end{cases}$$

so

$$\frac{\partial}{\partial p_j}\left(\frac{\pi_p(N)}{N}\right) = \left(1 - \sum_{r=1}^{j} p_r\right)^{N-1} - \sum_{s=j}^{K-1} p_s (N-1)\left(1 - \sum_{r=1}^{s} p_r\right)^{N-2}.$$

If the left-hand side of this equation is 0, we get

$$(N-1)\sum_{s=j}^{K-1} p_s \left(1 - \sum_{r=1}^{s} p_r\right)^{N-2} = \left(1 - \sum_{r=1}^{j} p_r\right)^{N-1}. \quad (20)$$

We now prove (19) by induction on i. For $i = 1$, we set $j = K-1$ in (20) to get

$$(N-1)p_{K-1}\left(1 - \sum_{r=1}^{K-1} p_r\right)^{N-2} = \left(1 - \sum_{r=1}^{K-1} p_r\right)^{N-1}$$

so $Np_{K-1} = 1 - \sum_{r=1}^{K-2} p_r$; i.e., the lemma is true for $i = 1$, since $f_1(N) = 0$.

Assume now that the lemma is true for $i = \ell$, so

$$(N - f_\ell(N))p_{K-\ell} = (1 - f_\ell(N))\left(1 - \sum_{r=1}^{K-(\ell+1)} p_r\right) \quad (21)$$

and

$$(N-1)p_{K-\ell} = (1 - f_\ell(N))\left(1 - \sum_{r=1}^{K-\ell} p_r\right). \quad (22)$$

Setting $j = K - (\ell+1)$ in (20) gives

$$\begin{aligned}(N-1)p_{K-(\ell+1)}&\left(1 - \sum_{r=1}^{K-(\ell+1)} p_r\right)^{N-2} \\ &+ (N-1)\sum_{s=K-\ell}^{K-1} p_s \left(1 - \sum_{r=1}^{s} p_r\right)^{N-2} \\ &= \left(1 - \sum_{r=1}^{K-(\ell+1)} p_r\right)^{N-1}\end{aligned}$$

so, by (20)

$$(N-1)p_{K-(\ell+1)}\left(1 - \sum_{r=1}^{K-(\ell+1)} p_r\right)^{N-2} + \left(1 - \sum_{r=1}^{K-\ell} p_r\right)^{N-1} = \left(1 - \sum_{r=1}^{K-(\ell+1)} p_r\right)^{N-1}.$$

It follows from (21) and (22) that

$$\begin{aligned}(N-1)p_{K-(\ell+1)}&\left(\frac{N - f_\ell(N)}{1 - f_\ell(N)}\right)^{N-2} \\ &+ \left(\frac{N-1}{1 - f_\ell(N)}\right)^{N-1} p_{K-\ell} \\ &= \left(\frac{N - f_\ell(N)}{1 - f_\ell(N)}\right)^{N-1} p_{K-\ell}\end{aligned}$$

and, by Definition 2

$$(N-1)p_{K-(\ell+1)+f_{\ell+1}(N)\frac{N-f_\ell(N)}{1-f_\ell(N)}p_{K-\ell}} = \frac{N - f_\ell(N)}{1 - f_\ell(N)} p_{K-\ell}.$$

This and (21) imply

$$(N-1)p_{K-(\ell+1)} = (1 - f_{\ell+1}(N))\left(1 - \sum_{r=1}^{K-(\ell+1)} p_r\right) \quad (23)$$

equivalently

$$(N - f_{\ell+1}(N))p_{K-(\ell+1)} = (1 - f_{\ell+1}(N))\left(1 - \sum_{r=1}^{K-(\ell+2)} p_r\right)$$

i.e., (19) is true for $i = \ell + 1$, completing the induction. $\square$

REFERENCES

[1] D. Bertsekas and R. Gallager, *Data Networks*, 2nd ed. New York: Prentice-Hall, 1987.

[2] L. Kleinrock and F. Tobagi, "Packet switching in radio channels: Part I—Carrier sense multiple access modes and their throughput-delay characteristics," *IEEE Trans. Commun.*, vol. 23, pp. 1400–1416, Dec. 1975.

[3] V. Paxson and S. Floyd, "Wide area traffic: The failure of poisson modeling," in *IEEE/ACM Trans. Networking*, vol. 3, June 1995, pp. 226–244.
[4] B. Chen, K. Jamieson, H. Balakrishnan, and R. Morris, "Span: An energy-efficient coordination algorithm for topology maintainence in ad hoc wireless networks," in *Proc. 7th Int. ACM Conf. Mobile Computing Networking (MOBICOM)*, Rome, Italy, July 2001, pp. 85–96.
[5] N. Priyantha, A. Chakraborty, and H. Balakrishnan, "The cricket location-support system," in *Proc. 6th Int. ACM Conf. Mobile Computing Networking (MOBICOM)*, Boston, MA, Aug. 2000, pp. 32–43.
[6] J. Kahn, R. Katz, and K. Pister, "Mobile networking for smart dust," in *Proc. 5th Int. ACM Conf. Mobile Computing Networking (MOBICOM)*, Seattle, WA, Aug. 1999, pp. 271–278.
[7] D.-M. Chiu and R. Jain, "Analysis of the increase/decrease algorithms for congestion avoidance in computer networks," *Comput. Networks ISDN Syst.*, vol. 17, no. 1, pp. 1–14, June 1989.
[8] K. Jamieson, H. Balakrishnan, and Y. C. Tay. (2003, May) Sift: A MAC protocol for event-driven wireless sensor networks. MIT Lab. Comput. Sci., Tech. Rep. 894. [Online]. Available: http://www.lcs.mit.edu/publications/pubs/pdf/MIT-LCS-TR-894.pdf
[9] Y. Xu, J. Heidemann, and D. Estrin, "Geography-informed energy conservation for ad hoc routing," in *Proc. 7th Int. ACM Conf. Mobile Computing Networking (MOBICOM)*, Rome, Italy, 2001, pp. 70–84.
[10] W. Ye, J. Heidemann, and D. Estrin, "An energy-efficient MAC protocol for wireless sensor networks," in *Proc. IEEE INFOCOM*, New York, June 2002, pp. 1567–1576.
[11] S. Singh, M. Woo, and C. S. Raghavendra, "Power-aware routing in mobile ad hoc networks," in *Proc. 5th Annu. ACM/IEEE Int. Conf. Mobile Computing Networking (MOBICOM)*, Dallas, TX, Oct. 1998, pp. 181–190.
[12] *Wireless LAN Medium Access Control and Physical Layer Specifications*, Aug. 1999. IEEE 802.11 Standard.
[13] S. McCanne and S. Floyd. Ns Notes and Documentation. [Online]. Available: http://www.isi.edu/vint/nsnam/
[14] C.-Y. Wan, S. Eisenmen, and A. Campbell, "CODA: Congestion control in sensor networks," in *Proc. ACM Sensys Conf.*, Los Angeles, CA, Nov. 2003, pp. 266–279.
[15] A. Woo and D. Culler, "A transmission control scheme for media access in sensor networks," in *Proc. 7th Intl. ACM Conf. Mobile Computing Networking (MOBICOM)*, Rome, Italy, July 2001, pp. 221–235.
[16] V. Bharghavan, "MACAW: A media access protocol for wireless LANs," in *Proc. ACM Conf. Applications, Technologies, Architectures, Protocols for Computer Communication (SIGCOMM)*, London, U.K., Aug. 1994, pp. 212–225.
[17] J. Gao and I. Rubin, "Analysis of random access protocol under bursty traffic," presented at the Proc. 4th IFIP/IEEE Int. Conf. Management Multimedia Networks Services, Chicago, IL, Oct. 2001.
[18] R. Metcalfe and D. Boggs, "Ethernet: Distributed packet switching for local computer networks," in *Commun. ACM*, vol. 19, July 1976, pp. 395–404.
[19] F. Calì, M. Conti, and E. Gregori, "Dynamic tuning of the IEEE 802.1 1 protocol to achieve a theoretical performance limit," in *IEEE/ACM Trans. Networking*, vol. 8, Dec. 2000, pp. 785–799.
[20] Z. Cai, M. Lu, and X. Wang, "Randomized broadcast channel access algorithms for ad hoc networks," in *Proc. IEEE Int. Conf. Parallel Processing*, Aug. 2002, pp. 151–158.
[21] *High Performance Radio Local Area Network (HIPERLAN) Type 1: Functional Specification*, 1996. European Telecommunication Standard.
[22] K.-O. Cho, H.-C. Shin, and J.-K. Lee, "Performance analysis of HIPERLAN channel access control protocol," in *Proc. ICICE Trans. Commun.*, vol. E85-B, 2002, pp. 2044–2052.
[23] C. Fullmer and J. Garcia-Luna-Aceves, "Floor acquisition multiple-access (FAMA) for packet radio networks," in *Proc. ACM Conf. Applications. Technologies, Architectures, Protocols Computer Communication (SIGCOMM)*, Cambridge, MA, Aug. 1995, pp. 262–273.
[24] R. Garcés and J. J. Garcia-Luna-Aceves, "Floor acquisition multiple access with collision resolution," in *Proc. 2nd Int. ACM Conf. Mobile Computing Networking (MOBICOM)*, Rye, NY, Nov. 1996, pp. 187–197.
[25] F. Tobagi, "Multiaccess protocols in packet communication systems," *IEEE Trans. Commun.*, vol. 28, pp. 468–488, Apr. 1980.
[26] W. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "An application-specific protocol architecture for wireless microsensor networks," *IEEE Trans. Wireless Commun.*, vol. 1, pp. 660–670, Oct. 2002.
[27] O. Mowafi and A. Ephremides, "Analysis of a hybrid access scheme for buffered users—Probabilistic time division," *IEEE Trans. Software Eng.*, vol. 8, pp. 52–60, Jan. 1982.
[28] I. Chlamtac and A. Faragó, "Making transmission schedules immune to topology changes in multihop packet radio networks," in *IEEE/ACM Trans. Networking*, vol. 2, Apr. 1994, pp. 23–29.
[29] I. Chlamtac, A. Faragó, and H. Zhang, "Time-spread multiple-access (TSMA) protocols for multihop mobile radio networks," in *IEEE/ACM Trans. Networking*, vol. 5, Dec. 1997, pp. 804–812.

Y. C. Tay received the B.Sc. degree from the University of Singapore and Ph.D. degree from Harvard University, Cambridge, MA.

He has a joint appointment with the Departments of Mathematics and Computer Science, National University of Singapore (http://www.comp.nus.edu.sg/~tayyc). His main research interest is performance modeling (transaction processing, multimedia load-sharing, wireless access, parallel rendering, page replacement, and Internet equilibrium). Other interests include distributed protocols and their correctness proofs.

Kyle Jamieson received the B.S. degree in mathematics, the B.S. degree in computer science, and the M.Eng. degree in computer science from the Massachusetts Institute of Technology (MIT), Cambridge, MA, in 2000, 2001, and 2001, respectively. He is currently working toward the Ph.D. degree in the Networks and Mobile Systems Group, Computer Science and Artificial Intelligence Laboratory, MIT.

His research interests are in wireless and sensor systems, focusing on energy-efficiency, medium access control, and congestion control.

Hari Balakrishnan received the B.Tech. degree from the Indian Institute of Technology, Madras, in 1993 and the Ph.D. degree in computer science from the University of California at Berkeley, in 1998.

He is an Associate Professor in the Department of Electrical Engineering and Computer Science and a Member of the Computer Science and Artificial Intelligence Laboratory (CSAIL), Massachusetts Institute of Technology (MIT), Cambridge, MA. He leads the Networks and Mobile Systems Group at CSAIL, which explores research issues in computer networks, wireless networking, mobile systems, and pervasive computing.

Prof. Balakrishnan is the recipient of an Alfred P. Sloan Research Fellowship (2002), a National Science Foundation CAREER Award (2000), the ACM Doctoral Dissertation Award for his work on reliable data transport over wireless networks (1998), and several best paper awards at technical conferences. He was awarded the Ruth and Joel Spira Award for Distinguished Teaching in 2001, the Junior Bose Award for Excellence in Teaching in 2002 from the MIT School of Engineering, and MIT's Edgerton Faculty Achievement Award for Distinction in Research and Teaching in 2003.