State Dependent Attempt Rate Modeling of Single Cell IEEE 802.11 WLANs with Homogeneous Nodes and Poisson Arrivals

Manoj K. Panda and Anurag Kumar
ECE Department, Indian Institute of Science, Bangalore -560012. Email: {manoj,anurag}@ece.iisc.ernet.in

***Abstract*—Analytical models of IEEE 802.11-based WLANs are invariably based on approximations, such as the well-known *mean-field* approximations proposed by Bianchi for saturated nodes. In this paper, we provide a new approach for modeling the situation when the nodes are not saturated. We study a State Dependent Attempt Rate (SDAR) approximation to model M queues (one queue per node) served by the CSMA/CA protocol as standardized in the IEEE 802.11 DCF. The approximation is that, when n of the M queues are non-empty, the attempt probability of the n non-empty nodes is given by the long-term attempt probability of n saturated nodes as provided by Bianchi's model. This yields a coupled queue system. When packets arrive to the M queues according to independent Poisson processes, we provide an exact model for the coupled queue system with SDAR service. *The main contribution of this paper is to provide an analysis of the coupled queue process by studying a lower dimensional process and by introducing a certain conditional independence approximation*. We show that the numerical results obtained from our finite buffer analysis are in excellent agreement with the corresponding results obtained from *ns-2* simulations. We replace the CSMA/CA protocol as implemented in the *ns-2* simulator with the SDAR service model to show that the SDAR approximation provides an accurate model for the CSMA/CA protocol. We also report the simulation speed-ups thus obtained by our *model-based simulation*.**

***Index Terms*—IEEE 802.11, non-saturated analysis, throughput and delay modeling, state dependent attempt rate, coupled queue analysis, state space reduction**

I. Introduction

Analytical modeling of Wireless Local Area Networks (WLANs) based on the IEEE 802.11 standard [1] has been a topic of great interest in the past few years. In his seminal work, Bianchi [2] proposed an accurate model under *saturated* conditions[1]. The key approximations in Bianchi's model are: (B1) in a randomly chosen slot, each node attempts with a constant probability β independent of all the other nodes, and (B2) every attempt collides with a constant probability γ regardless of the number of collisions already suffered. As conjectured by Bianchi, the "independence approximation" (B1) of his model is exact under saturated conditions in the *mean-field* limit, a formal proof of which can be found in [3]. Cali et. al [4] proposed and analyzed a p-persistent version of IEEE 802.11. Kumar et al. [5] generalized the Bianchi model to arbitrary back-off multipliers and retry limits.

[1]A wireless node is said to be saturated if it is perpetually backlogged, i.e., if the node always contains one or more packets awaiting transmission such that its transmission queue never becomes empty.

The saturation assumption (as taken in [2], [4] and [5]) is not valid for many real applications (e.g., web, email, VoIP). Consequently, modeling of the non-saturated case attracted much attention. Important contributions in this direction include [6], [7], [8], [9], [10], [11] for Poisson traffic; [12] for TCP transfers; and [13] for TCP transfers as well as VoIP sessions. The models in [6]-[10] retain Approximations B1 and B2 of the Bianchi model. The impact of traffic intensity is modeled by the "probability of a queue being empty" which, in turn, is obtained by queueing approximations. In [11], the authors argue that Approximation B1 of the Bianchi model leads to inaccurate results under non-saturated conditions. They propose to model the attempt probability as a function of the number of non-empty nodes in the system. Essentially, instead of using a constant attempt probability β for all the states of the system, they use a set of *state dependent* attempt probabilities β_n's where the state of the system in a given time slot is the number of non-empty nodes n in that time slot.

To our knowledge, the model proposed in [11] is by far the most accurate under the Poisson traffic assumption and yields results that match extremely well with *ns-2* simulations. We emphasize that the most referred work by Tickoo and Sikdar [9] is actually inaccurate in predicting the collision probabilities and the packet delays since they use Approximation B1 of the Bianchi model. We also remark that state dependent attempt probabilities have been successfully applied to accurately model TCP-controlled large file transfers over a WLAN as well as analyze the VoIP capacity of WLANs [12], [13]. The analytical models in [2]-[5], [7]-[13] deal with *single cell* WLANs whereas [6] proposes a model for multi-hop networks as well. Since multi-hop networks are beyond the scope of this paper, we shall not discuss them.

Comparison with Earlier Works: In this paper, we provide a new approach for modeling single cell WLANs under non-saturated conditions. Each node in the WLAN uses the IEEE 802.11 Distributed Coordination Function (DCF) Medium Access Control (MAC) protocol (also known as the CSMA/CA protocol) to schedule its transmissions. The details of the protocol can be found in [1] and [2]. We consider Poisson traffic arrivals and propose a model in the same spirit as that in [11], i.e., we model the attempt probability as a function of the number of non-empty nodes. The state dependent attempt probabilities in [11] are obtained by an iterative analysis which requires computations involving a three-dimensional Markov chain. We, however, apply the heuristic approximation of [13] to obtain the state dependent attempt probabilities (See Approximation 3.1 in Section III-B). As explained in Section IV-E, this heuristic makes our model computationally less expensive than that in [11]. In particular, our model requires computations involving a two-dimensional Markov chain. We emphasize that, even though we apply the heuristic of [13], the problem addressed in this paper is significantly different from that in [13]. The problem setting in [13] is such that the number of packets in a non-saturated queue never exceeds 1 so that analysis of the queue dynamics is not required. In this paper, we address a situation where the queues can grow as large as the buffer capacity and we analyze a coupled queue system. We show that the heuristic approximation of [13] is useful in our context as well.

Our Contributions: We make the following contributions:

- We apply a State Dependent Attempt Rate (SDAR) approximation to model the attempt processes of the nodes. This yields a coupled queue system. We provide an exact Markov model for the coupled queue system with Poisson arrivals and SDAR service.
- We propose a technique to reduce the state space of the coupled queue system and analyze the reduced state process for infinite as well as for finite buffer sizes. Our finite buffer analysis is computationally less expensive than that in [11], yet yields numerical results as accurate as those in [11].
- We apply the SDAR model to modify the *ns-2* simulator. Our objective in doing so is to improve the speed of simulation by a *model-based simulation* at the MAC layer. We show that the SDAR model of contention provides an accurate model for the CSMA/CA protocol and, at the same time, achieves speed-ups (w.r.t. MAC layer operations) up to 1.55 to 5.4 depending on the arrival rate and the number of nodes in the WLAN.

Brief Outline of the Paper: We summarize our modeling assumptions in Section II. In Section III we introduce the SDAR approximation and develop an exact Markov model for the coupled queue system. In Section IV we reduce the state space of the coupled queue system and develop the analysis for infinite and finite buffer sizes. Derivation of important performance measures is carried out in Section V. In Section VI we report how the SDAR heuristic technique could be effectively applied to improve simulation speed. In Section VII we validate our model by comparing with the results obtained from *ns-2* simulations. We conclude the paper in Section VIII. The Appendix contains a proof of the theorem that appears in Section III-B.

II. Modeling Assumptions

M.1 The WLAN consists of M *homogeneous* nodes, which means that the nodes use identical *DCF parameters*.

M.2 The arrival processes to the MAC queues bring fixed-length packets according to independent Poisson processes with rate λ packets/sec.

M.3 We consider single cells containing no hidden nodes.

M.4 Nodes cannot *capture* packets in the presence of interference due to simultaneous transmissions. This implies that: i) simultaneous transmissions in the WLAN always result in the corruption of packets of all the involved nodes, and ii) *there can be at most one successful transmission in the system at any point of time*.

M.5 The MAC queue of each node has infinite buffer space, i.e., packets are never dropped due to buffer overflow. (This assumption will be dropped in Section IV-D and we will develop an analysis for finite buffer sizes)

M.6 The wireless channel is error-free which implies that single transmissions in the WLAN are always successful.

III. The Model

Since by Assumption M.4 at most one packet can depart from the system at a time, the WLAN can be viewed as a single server serving multiple queues. The M queues corresponding to the M nodes in the WLAN are essentially coupled due to MAC contention and our immediate objective is to analyze the joint queue length process. We proceed by embedding the process at the so called channel slot boundaries described in the following.

A. Channel Slots

In IEEE 802.11, the random back-off periods are multiples of a time unit called the *back-off slot*. Let σ denote the duration of a back-off slot in seconds. We call a time unit equal to the duration σ of a back-off slot, a *system slot*. Henceforth, all time durations, i.e., times during back-off periods, “system empty” periods[2] and activity periods will be measured in terms of system slots. For analytical convenience, we make the following approximations:

A.1 Nodes always sample non-zero back-offs.

A.2 Time is slotted with slot length σ during the system empty periods.

In reality, nodes do sample 0 back-offs with some positive probability. But this possibility affects the final results only marginally. As a consequence of Approximation A.1, every activity (i.e., a successful transmission or a collision) in the system is followed by a back-off slot. Approximation A.2 amounts to saying that the system empty periods are integer multiples of system slots, which, in reality, may not be true. However, the error introduced by this quantization is negligible if $M\lambda\sigma \ll 1$, i.e., if the total number of packet arrivals into all the nodes per system slot is much smaller than 1[3].

Owing to Approximations A.1 and A.2 we observe that the channel activity evolves over cycles, which we call *channel slots*. Figure 1 depicts the back-off evolution and activities in a WLAN. The channel slots that occur on the common medium have also been shown and the channel slot boundaries have been indicated by arrows. When the system is empty, an idle channel slot of duration σ occurs (see Approximation A.2). A

[2] System empty periods are the times during which all the nodes in the WLAN have empty queues.

[3] We remark that the packet transmission times typically take hundreds of system slots. Hence, $M\lambda\sigma \ll 1$ holds whenever the queues are stable.

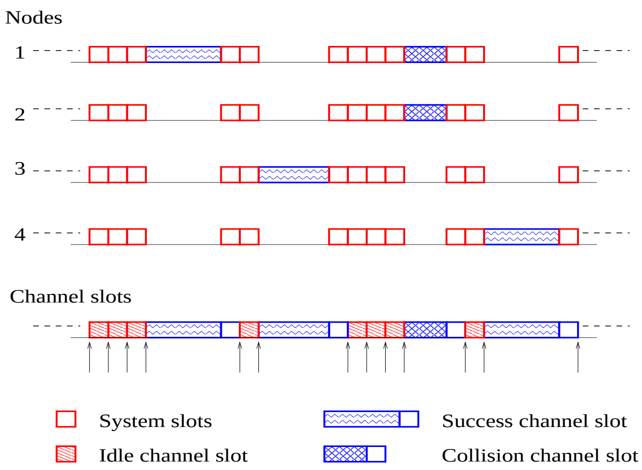

Fig. 1. Description of channel slots.

succession of idle channel slots occur until arrivals make some of the nodes non-empty. Non-empty nodes sample non-zero back-offs (see Approximation A.1) and attempt transmissions when their back-off counters become 0. Depending on whether there are no attempts, only one attempt, or more than one attempt made in the system, an idle, a success, or a collision channel slot occurs. The duration of an idle channel slot when the system is non-empty is equal to the duration σ of a back-off slot. By Approximation A.1 every activity is followed by a back-off slot. Hence, the duration of a success (resp. collision) channel slot is the *success time* T_s (resp. *collision time* T_c) plus a back-off slot where T_s and T_c are in system slots[4]. The attempt process resumes at the end of channel slots thereby creating more channel slots and the process repeats.

B. A Coupled Queue Formulation

We model the evolution of the system in discrete time with time epochs embedded at the channel slot boundaries. Figure 2 depicts the evolution of a typical node in the system. Let $T(t), t = 0, 1, 2, 3, \cdots\cdots$, with $T(0) = 0$, denote the channel slot boundaries which form a sequence $\{T(t), t \geq 0\}$ of random times. The t^{th} channel slot ($t \geq 1$), the duration of which we denote by $L(t)$, is precisely the time interval $[T(t-1), T(t))$. Henceforth, our discussion will be in terms of the discrete time index t. Note that the discrete time instants indexed by t correspond to the actual (i.e., continuous) time instants $T(t)$.

Let $Q_i(t), t \geq 0, i = 1, 2, \cdots, M$, denote the number (of packets) in the i^{th} node's MAC queue at time t. Let $A_i(t)$ (resp. $D_i(t)$), $t \geq 1, i = 1, 2, \cdots, M$, denote the number of arrivals into (resp. departures from) the i^{th} node's MAC queue in the t^{th} channel slot. Note that $\forall t \geq 0, i = 1, 2, \cdots, M$, we have $Q_i(t) \in \mathbb{N}$ where $\mathbb{N} := \{0, 1, 2, \cdots\}$; $\forall t \geq 1, i = 1, 2, \cdots, M$, we have $A_i(t) \in \mathbb{N}$ and $D_i(t) \in \{0, 1\}$. The last

[4]The success time T_s is the number of system slots taken for the transmission of an entire frame sequence (e.g. DATA-SIFS-ACK in the "basic access" mode or RTS-SIFS-CTS-SIFS-DATA-SIFS-ACK in the "RTS/CTS" mode) plus a DIFS. The collision time T_c is the number of system slots until the frame sequence gets aborted due to a *timeout* which includes an EIFS. Note that T_s and T_c need not be integer multiples of system slots.

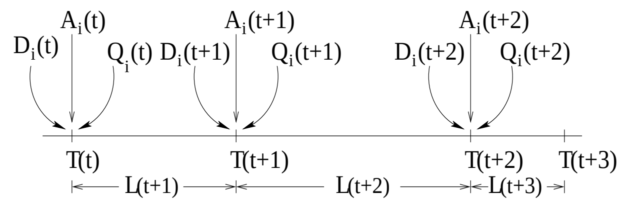

Fig. 2. Evolution of Node i's queue. Notice that $Q_i(t)$, $A_i(t)$ and $D_i(t)$ are embedded at times $T(t)_+$, $T(t)$ and $T(t)_-$, respectively.

constraint follows from the fact that there can be at most one success in a channel slot. Clearly, the "number in the queue" processes $\{Q_i(t), t \geq 0\}$ $(1 \leq i \leq M)$ evolve as (notice the embedding in Figure 2)

$$Q_i(t+1) = Q_i(t) - D_i(t+1) + A_i(t+1) \qquad (1)$$

with the convention that $Q_i(t) = 0 \Rightarrow D_i(t+1) = 0$.

Due to the "Poisson arrivals" assumption, the distribution of the number of arrivals in a channel slot depends only on the duration of the channel slot. The duration of a channel slot is known if the channel slot type (i.e., whether it is an idle, a success or a collision channel slot) is known (see Section III-A). Let L_{idle}, L_{succ} and L_{coll} denote the duration of an idle, a success and a collision channel slot, respectively. With slight abuse of notation, we indicate the occurrence of an idle, a success and a collision channel slot by $L(t) = L_{idle}$, $L(t) = L_{succ}$ and $L(t) = L_{coll}$, respectively, and define, $\forall t \geq 0, \forall j \geq 0, 1 \leq i \leq M$, the following probabilities:

$$\begin{aligned} d(j) &:= P\Big(A_i(t+1) = j \Big| L(t+1) = L_{idle}\Big) \\ s(j) &:= P\Big(A_i(t+1) = j \Big| L(t+1) = L_{succ}\Big) \\ c(j) &:= P\Big(A_i(t+1) = j \Big| L(t+1) = L_{coll}\Big) \end{aligned} \qquad (2)$$

The "Poisson arrivals" assumption also implies that the probabilities $d(j)$, $s(j)$ and $c(j)$ are obtainable from Poisson distributions with mean $\lambda L_{idle} = \lambda\sigma$, $\lambda L_{succ} = \lambda\sigma(1 + T_s)$ and $\lambda L_{coll} = \lambda\sigma(1 + T_c)$, respectively, where, as pointed out earlier, T_s and T_c are in system slots.

As described in Section III-A, nodes can attempt only at the channel slot boundaries. Only those nodes that are non-empty at time t can attempt at time t. Let $N(t)$ denote the number of non-empty nodes in the system at time t. Then, by definition,

$$N(t) = \sum_{i=1}^{M} \mathbf{1}_{\{Q_i(t)>0\}} \qquad (3)$$

where $\mathbf{1}_{\{.\}}$ denotes the indicator function. We now introduce an important approximation regarding the attempt processes of the nodes proposed in [13].

Approximation 3.1: At any generic channel slot boundary t, every non-empty node attempts a transmission with probability $\beta_{N(t)}$ where β_n is the long-term attempt rate of the nodes in a single cell containing n saturated nodes.

We call Approximation 3.1 the **State Dependent Attempt Rate (SDAR)** approximation. The β_n's in the SDAR approximation can be obtained from the model in [2] or [5]. As a consequence of the SDAR approximation, given $N(t) = n$, the number of attempts made in the system at the channel slot boundary t is binomially distributed with parameters n and β_n. Hence, the probabilities with which the $(t+1)^{th}$ channel slot is an idle, a success or a collision channel slot can be easily determined as follows:

$$\begin{aligned} p_{idle,n} &:= P\Big(L(t+1) = L_{idle}\big|N(t) = n\Big) = (1-\beta_n)^n \\ p_{succ,n} &:= P\Big(L(t+1) = L_{succ}\big|N(t) = n\Big) \\ &= n\beta_n(1-\beta_n)^{n-1} \\ p_{coll,n} &:= P\Big(L(t+1) = L_{coll}\big|N(t) = n\Big) \\ &= 1 - p_{idle,n} - p_{succ,n} \end{aligned} \quad (4)$$

Furthermore, in case of a success channel slot, the packet departure can occur from any of the non-empty queues with equal probability since the aggregate attempt process is binomial. Thus, the number of departures $D_i(t+1)$, $t \geq 0, 1 \leq i \leq M$, in the $(t+1)^{th}$ channel slot satisfy

$$P\left(D_i(t+1) = 1\big|N(t) = n, L(t+1) = L_{succ}, Q_i(t) > 0\right) = \frac{1}{n}$$

The joint queue length process $\{\boldsymbol{Q}(t), t \geq 0\}$, where

$$\boldsymbol{Q}(t) := \big(Q_1(t), Q_2(t), \cdots, Q_M(t)\big),$$

completely determines the dynamics of the system. Approximations A.1 and A.2 and the SDAR approximation imply that $\{\boldsymbol{Q}(t), T(t), t \geq 0\}$ is a Markov renewal sequence with the process $\{\boldsymbol{Q}(t), t \geq 0\}$ being the embedded Discrete Time Markov Chain (DTMC).

Theorem 3.1: The DTMC $\{\boldsymbol{Q}(t), t \geq 0\}$ is positive recurrent if

$$M\lambda < \min_{1 \leq n \leq M} \Theta_{sat,n}$$

where $\Theta_{sat,n}$ is the mean aggregate throughput in packets/sec in a WLAN consisting of n saturated nodes.

Proof: See Appendix A. ■

Remarks 3.1: In general, the variation of $\Theta_{sat,n}$ with n depends on the back-off parameters such as the back-off window sizes and the retry limit. However, for the back-off parameters as prescribed by the IEEE 802.11 standard, we observe that, for M large enough ($M \geq 5$ suffices), $\min_{1 \leq n \leq M} \Theta_{sat,n} = \Theta_{sat,M}$. Thus, the DTMC $\{\boldsymbol{Q}(t), t \geq 0\}$ is positive recurrent if the aggregate arrival rate $M\lambda$ is less than the aggregate throughput $\Theta_{sat,M}$ for M saturated nodes.

We define the stationary probabilities of $\{\boldsymbol{Q}(t), t \geq 0\}$ as

$$\nu(\boldsymbol{k}) := P\left(\boldsymbol{Q} = \boldsymbol{k}\right) = P\left(Q_1 = k_1, Q_2 = k_2, \cdots, Q_M = k_M\right)$$

where $\boldsymbol{k} = (k_1, k_2, \cdots, k_M) \in \mathbb{N}^M$. In Section V we show that important performance measures such as collision probability and throughput can be derived once we know the stationary distribution of the process $\{N(t), t \geq 0\}$. In principle, it is possible to obtain the stationary disribution of the process $\{N(t), t \geq 0\}$ from the stationary distribution of the DTMC $\{\boldsymbol{Q}(t), t \geq 0\}$ (see Equation 3). However, for a WLAN containing M nodes, the DTMC $\{\boldsymbol{Q}(t), t \geq 0\}$ has a state space $\mathcal{S}^{(M)} = \mathbb{N}^M$ which is difficult to handle for $M > 1$[5]. For $M > 1$, it is difficult to analytically solve the M-dimensional DTMC $\{\boldsymbol{Q}(t), t \geq 0\}$. Furthermore, it is impossible to numerically solve the M-dimensional DTMC $\{\boldsymbol{Q}(t), t \geq\}$ since the state space is infinite. Hence, we verify the accuracy of our model in the following two ways:

1. In Section IV we propose a state reduction technique and develop an analysis of the reduced state process for finite buffer sizes. In Section VII we show that the numerical results obtained from our finite buffer analysis can accurately predict the corresponnnding results obtained from *ns-2* simulations. This makes our model predictive for finite buffer sizes.
2. We replace the detailed CSMA/CA protocol of the IEEE 802.11 standard as implemented in *ns-2* with the simple SDAR model (see Section VI). In Section VII we compare the simulation results obtained from (a) the unmodified implementation in *ns-2* with (b) the SDAR model in *ns-2* to show that the simulation results obtained from (a) and (b) match extremely well for finite as well as infinite buffer sizes. In particular, this validates the SDAR approximation even for infinite buffer sizes.

IV. Reduction of the State Space

Since the nodes are identical, we consider the following alternative description of the system. We define the state of the system at a channel slot boundary t as $\mathcal{X}(t) := \big(Q_1(t), \mathcal{M}(t)\big)$ where $Q_1(t)$ denotes the number (of packets) in the MAC queue of a tagged node at t and $\mathcal{M}(t)$ denotes the number of nodes, other than the tagged node, that are non-empty at t. The state space has now reduced to $\mathbb{N} \times \{0, 1, \cdots, M-1\}$. Note that $\mathcal{M}(t)$ and $N(t)$ are related as $N(t) = \mathbf{1}_{\{Q_1(t)>0\}} + \mathcal{M}(t)$. We define, the stationary probabilities of the process $\{\mathcal{X}(t), t \geq 0\}$ as $\pi(j,k) := P\left(Q_1 = j, \mathcal{M} = k\right)$. The $\pi(j,k)$'s can be derived from the $\nu(\boldsymbol{k})$'s as follows:

$$\pi(j,k) = \sum_{\left\{\boldsymbol{k}\,:\,k_1 = j,\, \sum_{i=2}^{M} \mathbf{1}_{\{k_i>0\}} = k\right\}} \nu(\boldsymbol{k}) \quad (5)$$

A. Approximating $\{\mathcal{X}(t)\}$ with a Markov Process

It is important to emphasize that the process $\{\mathcal{X}(t), t \geq 0\}$ is not Markovian. Since the state description $\mathcal{X}(t)$ does not keep track of the queue lengths of the nodes, other than the tagged node, the probability that "a departure from a non-tagged node leaves the queue empty" cannot be modeled by $\mathcal{X}(t)$. Clearly, a departure leaves the queue empty if the following two events occur:

- E.1 At the beginning of the success channel slot a queue contains exactly one packet which departs, and
- E.2 The queue does not receive any packets in the success channel slot.

Event E.2 is modeled by the probability $s(0)$ (see Equation 2). However, with the state description $\mathcal{X}(t)$, event E.1 cannot

[5]The $M = 1$ case can be easily analyzed, for example, by the generating function approach.

be modeled for the non-tagged queues. In fact, the balance equations for the $\pi(j,k)$'s can be derived from the balance equations for the $\nu(\boldsymbol{k})$'s using Equation 5 and it can be shown that the balance equations for $\pi(j,k)$, $j \neq 0$, $0 \leq k \leq M-1$, still contain some terms involving the $\nu(\boldsymbol{k})$'s which cannot be expressed entirely in terms of the $\pi(j,k)$'s[6]. Had $\{\mathcal{X}(t), t \geq 0\}$ been Markovian, the balance equations for all the $\pi(j,k)$'s could have been expressed entirely in terms of the $\pi(j,k)$'s.

Let $Q_d(t)$ denote the number of packets in the non-tagged queue at the channel slot boundary t from which a departure occurs at the channel slot boundary $t+1$. Note that $Q_d(t)$ must be positive; otherwise, there cannot be a departure from the d^{th} queue at $t+1$. Given that $Q_1(t) = j, \mathcal{M}(t) = k$, which is maintained by the state $\mathcal{X}(t)$, to model the evolution of $\{\mathcal{M}(t), t \geq 0\}$, we need the probability

$$P(Q_d(t+1) = 0 \mid Q_1(t) = j, \mathcal{M}(t) = k, Q_d(t) > 0)$$

which is equal to

$$P(Q_d(t) = 1 \mid Q_1(t) = j, \mathcal{M}(t) = k, Q_d(t) > 0) \cdot s_d(0)$$

where $s_d(0)$ is the probability that the queue from which the departure occurs receives no arrivals in the success channel slot. Notice that $\{\mathcal{X}(t), t \geq 0\}$ is not Markovian precisely because the probability $P(Q_d(t) = 1 \mid Q_1(t) = j, \mathcal{M}(t) = k, Q_d(t) > 0)$ cannot be modeled with the state description $\mathcal{X}(t)$. To be able to model event E.1, we apply an approximation introduced in [14] in the context of ALOHA networks.

Approximation 4.1: (**Conditional Independence**)

$$\begin{aligned} & P(Q_d(t) = 1 \mid Q_1(t) = j, \mathcal{M}(t) = k, Q_d(t) > 0) \\ = \; & P(Q_d(t) = 1 \mid N(t), Q_d(t) > 0) \end{aligned}$$

Approximation 4.1 has also been applied in [11] and amounts to saying that at any channel slot boundary t, the probability that a non-tagged queue contains exactly one packet, given that it is non-empty at t, is independent of the **exact number of packets** in the other queues (specifically, the tagged queue) and depends only on whether the other queues are empty or non-empty.

We approximate the process $\{\mathcal{X}(t), t \geq 0\}$ by a Markov process $\{\tilde{\mathcal{X}}(t), t \geq 0\}$ which has the same state description as that of $\{\mathcal{X}(t), t \geq 0\}$[7] and models event E.1 by invoking Approximation 4.1. Applying Approximation 4.1, we have

$$\begin{aligned} & P(\tilde{Q}_l = 1 \mid \tilde{Q}_1 = i, \tilde{\mathcal{M}} = n, \tilde{Q}_l > 0) \qquad (2 \leq l \leq M) \\ = \; & P(\tilde{Q}_l = 1 \mid \tilde{N} = \mathbf{1}_{\{i>0\}} + n, \tilde{Q}_l > 0) \\ = \; & \begin{cases} P(\tilde{Q}_l = 1 \mid \tilde{N} = n, \tilde{Q}_l > 0) & i = 0 \\ P(\tilde{Q}_l = 1 \mid \tilde{N} = n+1, \tilde{Q}_l > 0) & i > 0 \end{cases} \end{aligned} \tag{6}$$

For the process $\{\tilde{\mathcal{X}}(t), t \geq 0\}$, we define $\tilde{q}_n$, $1 \leq n \leq M$, to be the stationary probability that a non-tagged queue contains exactly one packet at a channel slot boundary given that it is non-empty and that n nodes are non-empty at that channel slot boundary, i.e.,

$$\tilde{q}_n := P\left(\tilde{Q}_i = 1 \mid \tilde{Q}_i > 0, \tilde{N} = n\right) \qquad (2 \leq i \leq M) \tag{7}$$

With the $\tilde{q}_n$'s defined as above, the probability that "a departure from a non-tagged queue leaves the queue empty at the channel slot boundary $t+1$" is given by $\tilde{q}_{\tilde{N}(t)} s(0)$ which is the joint probability of the events E.1 and E.2. Clearly, by construction, the process $\{\tilde{\mathcal{X}}(t), t \geq 0\}$ is a DTMC embedded at the channel slot boundaries. However, the $\tilde{q}_n$'s are not known *a priori*. For now, interpret the $\tilde{q}_n$'s as unknown parameters of the process $\{\tilde{\mathcal{X}}(t), t \geq 0\}$ yet to be determined.

B. An Iterative Method of Solution

Let $\tilde{\pi}(j,k)$, $j \geq 0; 0 \leq k \leq M-1$, denote the stationary probabilities of the DTMC $\{\tilde{\mathcal{X}}(t), t \geq 0\}$ (assuming that they exist). The $\tilde{\pi}(j,k)$'s are, in general, different from the $\pi(j,k)$'s[8]. Given the $\tilde{\pi}(j,k)$'s, the parameters $\tilde{q}_n$'s can be obtained as follows:

$$\begin{aligned} \tilde{q}_n &= P\left(\tilde{Q}_1 = 1 \middle| \tilde{Q}_1 > 0, \tilde{N} = n\right) && \text{(by homogeneity)} \\ &= \frac{\tilde{\pi}(1, n-1)}{\sum_{j=1}^{\infty} \tilde{\pi}(j, n-1)} && \text{(by definition)} \end{aligned} \tag{8}$$

This suggests an iterative method of solution to obtain the $\tilde{\pi}(j,k)$'s. For a given λ, assuming some values for the $\tilde{q}_n$'s, the balance equations for the $\tilde{\pi}(j,k)$'s can be solved along with the normalization equation

$$\sum_{j=0}^{\infty} \sum_{k=0}^{M-1} \tilde{\pi}(j,k) = 1 \tag{9}$$

to *uniquely* determine the $\tilde{\pi}(j,k)$'s (provided that the DTMC $\{\tilde{\mathcal{X}}(t), t \geq 0\}$ is positive recurrent). Equation 8 then provides new estimates of the $\tilde{q}_n$'s which can be used to obtain new estimates of the $\tilde{\pi}(j,k)$'s. This iterative procedure should continue until the solution converges within some specified tolerance limit. The stationary probabilities of the process $\{\tilde{N}(t), t \geq 0\}$ can then be obtained from the converged values of the $\tilde{\pi}(j,k)$'s as follows:

$$\begin{aligned} \tilde{p}_n &:= P\left(\tilde{N} = n\right) \\ &= P\left(\tilde{Q}_1 = 0, \tilde{\mathcal{M}} = n\right) + P\left(\tilde{Q}_1 > 0, \tilde{\mathcal{M}} = n-1\right) \\ &= \tilde{\pi}(0,n) + \sum_{j=1}^{\infty} \tilde{\pi}(j, n-1) \end{aligned} \tag{10}$$

We regard the $\tilde{p}_n$'s as "estimates" of the stationary probabilities p_n's of the process $\{N(t), t \geq 0\}$. Once the p_n's are approximated with the $\tilde{p}_n$'s, important performance measures can be obtained using the derivations in Section V.

[6] The balance equations for $\pi(0,k)$, $0 \leq k \leq M-1$, can be obtained entirely in terms of the $\pi(j,k)$'s using homogeneity of the nodes.

[7] In the remainder of this paper, a random variable X defined for the process $\{\mathcal{X}(t), t \geq 0\}$ will be correspondingly denoted as $\tilde{X}$ for the process $\{\tilde{\mathcal{X}}(t), t \geq 0\}$. Similarly, for a quantity $\tilde{r}$ defined for the process $\{\tilde{\mathcal{X}}(t), t \geq 0\}$, there is an analogous quantity r for the process $\{\mathcal{X}(t), t \geq 0\}$.

[8] If the buffer size of each of the M queues was equal to 1 packet, we would have $q_n = 1$ for all $1 \leq n \leq M$ and Approximation 4.1 would have been exact for the process $\{\mathcal{X}(t), t \geq 0\}$. In that case, the processes $\{\mathcal{X}(t), t \geq 0\}$ and $\{\tilde{\mathcal{X}}(t), t \geq 0\}$ would have been one and the same and their stationary probabilities would have been identical.

C. Transition Probability Matrix of $\{\tilde{\mathcal{X}}(t)\}$

For $0 \le n, k \le M-1$, we define the transition probabilities of the DTMC $\{\tilde{\mathcal{X}}(t), t \ge 0\}$ as follows:

$A_j(n,k) :=$ the transition probability from the state $(0,n)$ to the state (j,k), $j = 0,1,2,\cdots$

$B_j(n,k) :=$ the transition probability from the state (i,n) $(i > 0)$ to the state $(i+j,k)$, $j = -1,0,1,\cdots$

$A_j(n,k), j = 0,1,2,\cdots$ and $B_j(n,k), j = -1,0,1,\cdots$ can be obtained (we skip the derivations due to space constraints) and separating the terms that contain $\tilde{q}_n$ from the terms that do not contain $\tilde{q}_n$, it can be shown that

$$\begin{aligned} A_j(n,k) &= A_j^{(0)}(n,k) + \tilde{q}_n A_j^{(1)}(n,k) \\ B_j(n,k) &= B_j^{(0)}(n,k) + \tilde{q}_{n+1} B_j^{(1)}(n,k) \end{aligned} \quad (11)$$

where $A_j^{(0)}(n,k)$, $A_j^{(1)}(n,k)$, $B_j^{(0)}(n,k)$ and $B_j^{(1)}(n,k)$ are given by Equation 12 (provided at the top of the next page). Let $\boldsymbol{A_j}^{(0)}$, $\boldsymbol{A_j}^{(1)}$, $\boldsymbol{B_j}^{(0)}$ and $\boldsymbol{B_j}^{(1)}$ denote the $M \times M$ matrices with their $(n,k)^{th}$ entries given by $A_j^{(0)}(n,k)$, $A_j^{(1)}(n,k)$, $B_j^{(0)}(n,k)$ and $B_j^{(1)}(n,k)$, respectively. Using this matrix notation, Equation 11 can be rewritten as

$$\boldsymbol{A_j} = \boldsymbol{A_j}^{(0)} + \boldsymbol{\Delta_{\tilde{q}_0}} \boldsymbol{A_j}^{(1)} \;\; ; \;\; \boldsymbol{B_j} = \boldsymbol{B_j}^{(0)} + \boldsymbol{\Delta_{\tilde{q}_1}} \boldsymbol{B_j}^{(1)} \quad (13)$$

where $\boldsymbol{\Delta_{\tilde{q}_0}} = diag\,(0, \tilde{q}_1, \cdots, \tilde{q}_{M-1})$ and $\boldsymbol{\Delta_{\tilde{q}_1}} = diag\,(\tilde{q}_1, \tilde{q}_2, \cdots, \tilde{q}_M)$ are $M \times M$ diagonal matrices. We define $\boldsymbol{\tilde{\pi}^{(j)}} := (\tilde{\pi}(j,0), \tilde{\pi}(j,1), \cdots \tilde{\pi}(j, M-1))$ for all $j \ge 0$, and $\boldsymbol{\tilde{\pi}} := (\boldsymbol{\tilde{\pi}^{(0)}}, \boldsymbol{\tilde{\pi}^{(1)}}, \boldsymbol{\tilde{\pi}^{(2)}}, \cdots\cdots)$. Using this notation, the balance equations for the $\tilde{\pi}(j,k)$'s can be written as

$$\boldsymbol{\tilde{\pi}} = \boldsymbol{\tilde{\pi} P} \quad (14)$$

where the transition probability matrix $\boldsymbol{P}$ can be seen to have the following $M/G/1$ *type* structure

$$\boldsymbol{P} = \begin{bmatrix} \boldsymbol{A_0} & \boldsymbol{A_1} & \boldsymbol{A_2} & \boldsymbol{A_3} & \cdots \\ \boldsymbol{B_{-1}} & \boldsymbol{B_0} & \boldsymbol{B_1} & \boldsymbol{B_2} & \cdots \\ \boldsymbol{0} & \boldsymbol{B_{-1}} & \boldsymbol{B_0} & \boldsymbol{B_1} & \cdots \\ \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{B_{-1}} & \boldsymbol{B_0} & \cdots \\ \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix} \quad (15)$$

D. The Finite Buffer Case

To be able to compute the $\tilde{p}_n$'s by the iterative method described in Section IV-B we now consider finite buffer sizes for which the finite set of balance equations can be numerically solved. Let the buffer size of each of the M queues be K packets. We redefine $\boldsymbol{\tilde{\pi}^{(j)}} := (\tilde{\pi}(j,0), \tilde{\pi}(j,1), \cdots \tilde{\pi}(j, M-1))$ for all $0 \le j \le K$, and $\boldsymbol{\tilde{\pi}} := (\boldsymbol{\tilde{\pi}^{(0)}}, \boldsymbol{\tilde{\pi}^{(1)}}, \boldsymbol{\tilde{\pi}^{(2)}}, \cdots, \boldsymbol{\tilde{\pi}^{(K)}})$. The transition probability matrix $\boldsymbol{P_K}$ is now given by

$$\boldsymbol{P_K} = \begin{bmatrix} \boldsymbol{A_0} & \boldsymbol{A_1} & \boldsymbol{A_2} & \cdots & \boldsymbol{A_{K-1}} & \sum_{j=K}^{\infty} \boldsymbol{A_j} \\ \boldsymbol{B_{-1}} & \boldsymbol{B_0} & \boldsymbol{B_1} & \cdots & \boldsymbol{B_{K-2}} & \sum_{j=K-1}^{\infty} \boldsymbol{B_j} \\ \boldsymbol{0} & \boldsymbol{B_{-1}} & \boldsymbol{B_0} & \cdots & \boldsymbol{B_{K-3}} & \sum_{j=K-2}^{\infty} \boldsymbol{B_j} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} & \cdots & \boldsymbol{B_{-1}} & \sum_{j=0}^{\infty} \boldsymbol{B_j} \end{bmatrix}$$

where the $\boldsymbol{A_j}$'s and the $\boldsymbol{B_j}$'s are as defined earlier for the infinite buffer case[9]. It can be easily shown that the finite-state DTMC $\{\tilde{\mathcal{X}}_K(t), t \ge 0\}$ is irreducible, and hence, positive recurrent, if and only if the $\tilde{q}_n$'s are positive. Hence, if the iteration starts with positive $\tilde{q}_n$'s, Equations 14 and 9 can be solved yielding unique positive solutions for the $\tilde{\pi}(j,k)$'s. Equation 8 then provides new estimates of the $\tilde{q}_n$'s which are also positive[10]. Note that, for a given λ, the $d(j)$'s, $c(j)$'s and $s(j)$'s are known for all $j \ge 0$ (see Equation 2) and the β_n's are obtainable from a saturation analysis. Hence, for a given λ, the matrices $\boldsymbol{A_j}^{(0)}$, $\boldsymbol{A_j}^{(1)}$ $(j \ge 0)$ and $\boldsymbol{B_j}^{(0)}$, $\boldsymbol{B_j}^{(1)}$ $(j \ge -1)$ are computed only once. Each iteration gives a new set of $\tilde{q}_n$'s. Thus, the matrices $\boldsymbol{\Delta_{\tilde{q}_0}}$ and $\boldsymbol{\Delta_{\tilde{q}_1}}$ get updated in each iteration resulting in updated $\boldsymbol{A_j}$'s and the $\boldsymbol{B_j}$'s (see Equation 13). This, in turn, updates the $\boldsymbol{P_K}$ resulting in a new set of balance equations which yields a new set of $\tilde{q}_n$'s and the iteration continues until the results converge.

E. Complexity of the Finite Buffer Model

The SDAR approximation has the following advantages:

1. It enables computation of the β_n's using the saturation analysis of [2] or [5]. Thus, in contrast to the model in [11], the β_n's in our model are independent of the arrival rates and of the average queue occupancies. They are computed only once in the beginning of the overall computation by a separate procedure and then used as given parameters in the iterative analysis described in Sections IV-B and IV-D.
2. The effects of the back-off parameters are effectively captured in the pre-computed β_n's. Hence, they are not considered when modeling the queue length processes. This decoupling enables us to eliminate the first dimension, namely, "the back-off stage of the nodes" in the three-dimensional Markov chain of [11] and our model requires computations involving a two-dimensional Markov chain.
3. Since the β_n's are computed independent of the arrival rate of the Poisson arrival processes, they need not be computed for each arrival rate when studying the effect of arrival rate on performance measures.

Thus, the SDAR approximation makes our model computationally less expensive than that in [11] as follows. The complexity of the model in [11] is $O(RKM)$ where we recall that R denotes the retry limit, K denotes the buffer size and M denotes the number of nodes (see [11]). The complexity of obtaining β_n, $1 \le n \le M$, by a separate procedure is $O(RM)$ and the complexity of our finite buffer model, given the β_n's, is $O(KM)$. Thus, the overall complexity of our finite buffer model is $O(KM) + O(RM)$ which is less than the complexity $O(RKM)$ of the finite buffer model of

[9]Notice that $\boldsymbol{P_K}$ has a $M/G/1/K$ *type* structure where the infinite sums only requires summing up probabilities of Poisson distributions which can be simplified by observing that $\sum_{j=k}^{\infty} d(j) = 1 - \sum_{j=0}^{k-1} d(j)$ and so on.

[10]The infinite sum in Equation 8 will be a finite sum for finite buffer sizes. Similarly, the infinite sums in Equations 9 and 10 become finite sums for finite buffer sizes. Further, $\boldsymbol{P}$ in Equation 14 must be replaced by $\boldsymbol{P_K}$.

$$
\begin{aligned}
A_j^{(0)}(n,k) &= \binom{M-n-1}{k-n}\left(p_{idle,n}\, d(j)\ \ (1-d(0))^{k-n}\, d(0)^{M-k-1} + p_{coll,n}\ c(j)\ \ \cdots + p_{succ,n}\ s(j)\ \ \cdots\right) \\
A_j^{(1)}(n,k) &= p_{succ,n}\ s(j)\,(1-s(0))^{k-n}\ s(0)^{M-k-1}\left(\binom{M-n}{k-n+1}(1-s(0)) - \binom{M-n-1}{k-n}\right) \\
B_j^{(0)}(n,k) &= \binom{M-n-1}{k-n}\left(p_{idle,n+1}\ d(j)\ \ (1-d(0))^{k-n}\, d(0)^{M-k-1} + p_{coll,n+1}\ c(j)\ \ \cdots + p_{succ,n+1}\ s(j)\ \ \cdots\right) \\
&\quad + \binom{M-n-1}{k-n}\frac{p_{succ,n+1}}{n+1}\,(1-s(0))^{k-n}\, s(0)^{M-k-1}\,(s(j+1)-s(j)) \\
B_j^{(1)}(n,k) &= \left(\frac{n}{n+1}\right) p_{succ,n+1}\ s(j)\ \ (1-s(0))^{k-n}\ s(0)^{M-k-1}\left(\binom{M-n}{k-n+1}(1-s(0)) - \binom{M-n-1}{k-n}\right)
\end{aligned}
\tag{12}
$$

---

[11] for $R \geq 2, K \geq 2$. If one needs to solve for, say, l different arrival rates, to examine how the protocol behaves with the variation of traffic intensity, then the complexity of our model is $O(lKM) + O(RM)$ since we compute the β_n's only once and use them for all the l arrival rates whereas the complexity of the model in [11] for l different arrival rates is $O(lRKM)$. Thus, to study the effect of arrival rates on the performance measures, our model is far superior to that in [11]. This reduction in complexity is achieved precisely due to the SDAR approximation which does not require computing the attempt probabilities for each arrival rate.

V. Derivation of Performance Measures

As pointed out earlier, the processes $\{\boldsymbol{Q}(t), t \geq 0\}$ and $\{\tilde{\mathcal{X}}(t), t \geq 0\}$ (and their finite buffer versions) are DTMCs embedded at the channel slot boundaries $\{T(t), t \geq 0\}$. It is easy to see that $\{(\boldsymbol{Q}(t), T(t)), t \geq 0\}$ and $\{(\tilde{\mathcal{X}}(t), T(t)), t \geq 0\}$ are Markov renewal sequences. In this section, we apply Markov regenerative analysis to derive the important performance measures as follows.

Collision Probability: Let $\mathcal{A}(t)$ and $\mathcal{C}(t)$ denote the total number of attempts and collisions, respectively, up to time t where we recall that t is the discrete time index. Then the (conditional) collision probability γ is given by

$$
\gamma := \lim_{t\to\infty} \frac{\mathcal{C}(t)}{\mathcal{A}(t)} \stackrel{a.s.}{=} \frac{\sum_{n=0}^{M} p_n E_n C}{\sum_{n=0}^{M} p_n E_n A} \approx \frac{\sum_{n=0}^{M} \tilde{p}_n E_n C}{\sum_{n=0}^{M} \tilde{p}_n E_n A}
$$

where $E_n A$ and $E_n C$ denote the mean number of attempts and collisions *per channel slot* given that n nodes are non-empty. It is easy to see that $E_n A = n\beta_n$ and $E_n C = n\beta_n\left(1-(1-\beta_n)^{n-1}\right)$.

Throughput: Let $\mathcal{S}(t)$ denote the total number of successes up to time t. Then the aggregate system throughput Θ in packets/sec is given by

$$
\Theta := \lim_{t\to\infty} \frac{\mathcal{S}(t)}{t} \stackrel{a.s.}{=} \frac{\sum_{n=0}^{M} p_n E_n S}{\sum_{n=0}^{M} p_n E_n L} \approx \frac{\sum_{n=0}^{M} \tilde{p}_n E_n S}{\sum_{n=0}^{M} \tilde{p}_n E_n L}
$$

where $E_n S$ denotes the mean number of successes per channel slot and $E_n L$ denotes the mean channel slot duration in seconds (given that n nodes are non-empty). Again, it is easy to see that $E_n S = n\beta_n(1-\beta_n)^{n-1}$ and $E_n L = \sigma(1 + p_{coll,n}T_c + p_{succ,n}T_s)$. Further, due to homogeneity it follows that the per-node throughput $\theta = \frac{\Theta}{M}$.

Mean Packet Delay: We obtain the mean packet delay for our finite buffer model by applying a method proposed in [15] in the context of $M/G/1/K$ queues. We define the following:

$\alpha(j) :=$ fraction of time that the tagged queue contains j packets $(0 \leq j \leq K)$

$\pi^{(d)}(j) :=$ probability that a departure from the tagged queue leaves j packets behind $(0 \leq j \leq K-1)$

$\pi^{(a)}(j) :=$ probability that a packet accepted into the tagged queue finds j packets $(0 \leq j \leq K-1)$

Noting that departures can occur only at the end of channel slots, i.e., just before the the channel slot boundaries, $\pi^{(d)}(j)$, $0 \leq j \leq K-1$, can be obtained as follows:

$$
\begin{aligned}
\pi^{(d)}(j) &= \lim_{\tau\to\infty} \frac{\sum_{t=0}^{\tau} \mathbf{1}_{\{D_1(t+1)=1\}}\mathbf{1}_{\{Q_1(t+1)=j\}}}{\sum_{t=0}^{\tau} \mathbf{1}_{\{D_1(t+1)=1\}}} \\
&= \frac{\lim_{\tau\to\infty} \frac{1}{\tau}\sum_{t=0}^{\tau} \mathbf{1}_{\{D_1(t+1)=1\}}\mathbf{1}_{\{Q_1(t+1)=j\}}}{\lim_{\tau\to\infty} \frac{1}{\tau}\sum_{t=0}^{\tau} \mathbf{1}_{\{D_1(t+1)=1\}}}
\end{aligned}
\tag{16}
$$

For the SDAR model with buffer size K, it can be shown that, for $0 \leq j \leq K-2$, Equation 16 reduces to

$$
\pi^{(d)}(j) = \frac{\sum_{i=1}^{j+1}\sum_{n=0}^{M-1} \pi(i,n)\frac{p_{succ,n+1}}{n+1} s(j-i+1)}{\sum_{i=1}^{j+1}\sum_{n=0}^{M-1} \pi(i,n)\frac{p_{succ,n+1}}{n+1}}
$$

and for $j = K-1$, it reduces to

$$
\pi^{(d)}(K-1) = \frac{\sum_{i=1}^{K}\sum_{n=0}^{M-1} \pi(i,n)\frac{p_{succ,n+1}}{n+1}\left(1-\sum_{m=0}^{K-i-1} s(m)\right)}{\sum_{i=1}^{K}\sum_{n=0}^{M-1} \pi(i,n)\frac{p_{succ,n+1}}{n+1}}
$$

where the $s(j)$'s are given by Equation 2 and the $\pi(i,n)$'s can be approximated by $\tilde{\pi}(i,n)$. Note that the probability that an arrival is blocked is given by $\alpha(K)$. The mean rate at which packets are accepted into the queue must be equal to the mean rate at which packets depart from the queue. Hence, $\alpha(K)$ can be obtained by solving

$$
\lambda(1-\alpha(K)) = \frac{\Theta}{M} = \theta
$$

where Θ can be computed as explained earlier in this section. According to [15], we have

$$
\alpha(j) = \pi^{(a)}(j)(1-\alpha(K)), \quad 0 \leq j \leq K-1
$$

Since arrivals and departures occur one at a time, a level crossing analysis gives $\pi^{(a)}(j) = \pi^{(d)}(j)$, $0 \le j \le K-1$, and $\alpha(j)$, $0 \le j \le K-1$, can be obtained. The mean queue length $\bar{N}$ is then given by $\bar{N} = \sum_{j=0}^{K} j\alpha(j)$ from which the mean sojourn time or the mean packet delay $\bar{W}$ can be obtained as $\bar{W} = \frac{\bar{N}}{\theta}$.

VI. SDAR APPROXIMATION IN THE *ns-2* SIMULATOR

Wireless network simulators invariably employ simple models at the PHY layer to keep the simulations reasonably fast. In this section, we describe a model-based simulation technique at the MAC layer which is based on the SDAR attempt model. We applied the SDAR model in *ns-2* as follows. The simulator was modified to keep track of the number of non-empty nodes. Arrivals that occur during any activity (success or collision) are not taken into account until the activity finishes. Hence, the number of non-empty nodes does not change during the activities. Whenever an activity finishes or an arrival occurs during channel idle periods, the number of non-empty nodes is updated. Whenever the number of non-empty nodes is updated, all previously scheduled transmissions (if any) are canceled and random back-offs are sampled for each non-empty node using independent geometric random variables each having a mean $\frac{1}{\beta_n}$ where n denotes the current value of the number of non-empty nodes. Note that the geometric back-off durations with mean $\frac{1}{\beta_n}$ (which is equivalent to Bernoulli attempt processes with probabilities β_n) are easily obtained if the β_n's are known. The β_n's are pre-computed using the model in [5] and stored in a look-up table.

From the sampled back-offs, it is easy to determine which node(s) sampled the minimum back-off. If only one node samples the minimum back-off, the next event is a success. If two or more nodes sample the same minimum back-off, the next event is a collision. The appropriate event is then scheduled. Unless arrivals occur to empty queues before the "beginning of the scheduled event" epoch to increase the number of non-empty nodes (in which case the scheduled event is canceled), the scheduler clock is moved to the "end of the scheduled event" epoch. In case of a success, the DATA frame is handed over to the destination's MAC layer which then generates the corresponding ACK frame. These modifications have enabled us to achieve speed-ups up to 5.4. These speed-ups have been achieved with respect to the MAC layer operations and are summarized in Tables I and II for two different machines.

The observed speed-ups are obtained due to the following reasons. In *ns-2,* one transmission event per non-empty node remains pending in the "scheduler queue" and n back-off timers are kept running for n non-empty nodes. When a timer expires resulting in a transmission, all the other $n-1$ timers remain "paused" until the transmission finishes. When the timers resume, the remaining $n-1$ transmission events have to be rescheduled. Similar pausing and resheduling occurs when multiple timers expire simultaneously resulting in collisions. In our modifications, due to the memoryless property of the SDAR attempt model, we do not have to keep the back-offs sampled by the nodes. Moreover, at any point of time, only one event remains pending at the MAC layer which is the next event to occur on the common channel. This single pending event is interrupted and is rescheduled only if an arrival increases the number of non-empty nodes. Hence, the speed-up increases with the number of nodes M. The speed-up also increases with arrival rate λ since the average number of non-empty nodes increases with increase in λ. However, above a certain λ, the rate of cancellation of already scheduled events dominates and speed-up actually decreases with λ. The speed-up becomes constant beyond saturation. These observations are supported by the data in Tables I and II where the last row for each M corresponds to saturation.

The increase in speed-up with M is particularly desirable since *ns-2* is found to become worse with increase in M than λ. Also, note in Tables I and II that the speed-ups are more for the faster machine, i.e., MACHINE-I. This indicates that the speed-ups are not due to the incapability of the machines.

TABLE I
SPEED-UP FOR MACHINE-I (PENTIUM DUAL CORE, 2.80 GHZ, 1024 KB CACHE)

M	λ (pkts/sec)	MAC time *ns-2* (sec)	MAC time SDAR (sec)	Speed-up
50	5	45.045	9.405	4.79
50	10	103.43	19.16	5.4
50	15	158.44	39.40	4.02
30	5	10.8	3.25	3.08
30	10	22.88	6.89	3.32
30	15	32.49	10.24	3.17
30	20	60.49	22.84	2.65
10	10	2.233	0.943	2.37
10	20	4.314	1.864	2.31
10	30	7.256	3.114	2.33
10	40	8.351	3.641	2.30
10	50	10.777	4.767	2.26
10	60	13.269	5.789	2.30
10	70	19.922	10.312	1.93

TABLE II
SPEED-UP FOR MACHINE-II (PENTIUM, 1500 MHZ, 256 KB CACHE)

M	λ (pkts/sec)	MAC time *ns-2* (sec)	MAC time SDAR (sec)	Speed-up
50	5	102.06	31.99	3.19
50	10	268.23	66.86	4.0
50	15	369.13	97.77	3.78
30	5	23.9	9.59	2.49
30	10	48.16	19.06	2.53
30	15	75.93	29.79	2.55
30	20	136.82	54.11	2.53
10	10	3.66	2.13	1.72
10	20	7.31	4.25	1.72
10	30	11.38	6.62	1.72
10	40	15.52	9.00	1.72
10	50	20.25	11.28	1.795
10	60	25.87	13.99	1.85
10	70	31.07	20.02	1.55

VII. RESULTS AND DISCUSSION

In this section we compare analytical and simulation results to validate our model. The values of the DCF parameters in the *ns-2* simulator were taken as per the 802.11b standard. We

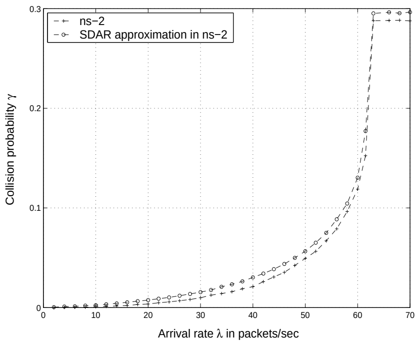

Fig. 3. Comparison of Collision Probability γ with infinite buffer size.

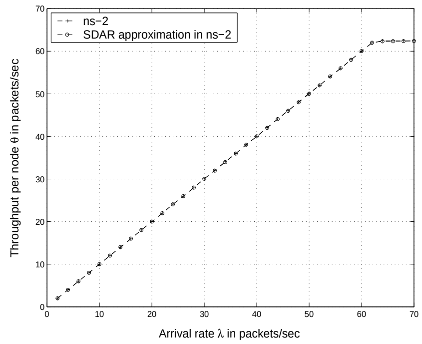

Fig. 4. Comparison of Throuput per Node θ with infinite buffer size.

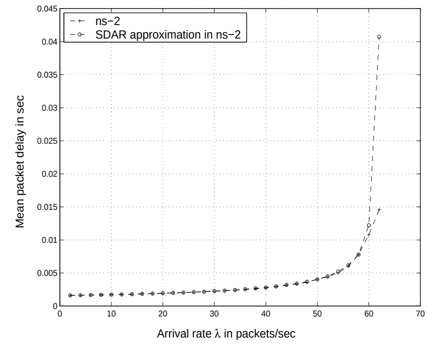

Fig. 5. Comparison of Mean Packet Delay $\bar{W}$ with infinite buffer size.

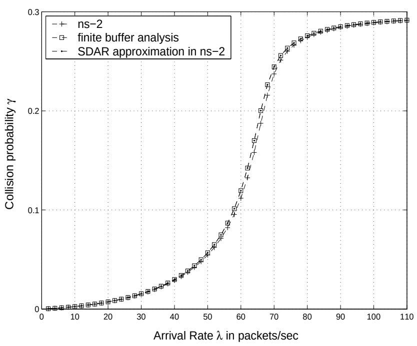

Fig. 6. Comparison of Collision Probability γ with finite buffer size $K = 5$.

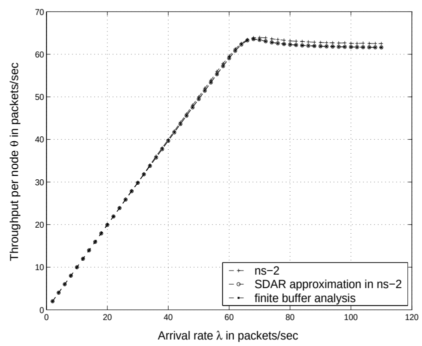

Fig. 7. Comparison of Throuput per Node θ with finite buffer size $K = 5$.

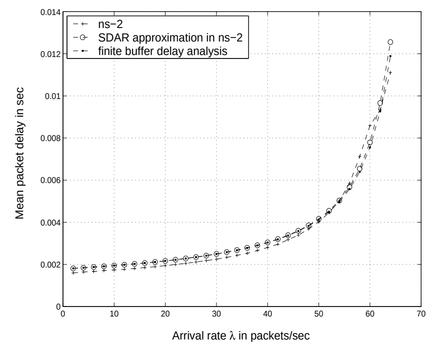

Fig. 8. Comparison of mean packet delay $\bar{W}$ with finite buffer size $K = 5$.

took Basic Rate = 2 Mbps, Data Rate = 11 Mbps, Packet Payload Size = 1000 bytes. We also make sure that Assumptions M.3, M.4 and M.6 are satisfied in our simulations. We provide results for the "basic access" case with $M = 10$ nodes. Similar results were obtained for the "RTS/CTS" case and other values of M. The analytical model was solved using MATLAB.

Figures 3, 4, and 5 compare the collision probability γ, the per-node throughput θ, and the mean packet delay $\bar{W}$ for the infinite buffer case. The buffer sizes in the *ns-2* simulator were set to very large values to simulate the infinite buffer assumption. It can be seen that the simulation results obtained from the unmodified *ns-2* and the SDAR approximation in *ns-2* match very well. The mismatch in the collision probabilities which lead to visible mismatch in the mean packet delays near saturation is mainly due to the over-estimation of collision probability by the Bianchi-type model of [5] which clearly appears as a $\sim 5\%$ mismatch of collision probability beyond the saturation threshold (which is about $\lambda = 62.5$ packets/sec for $M = 10$. See Figure 3). Note in Figure 4 that the saturation throughput is correctly predicted which remains constant beyond the saturation threshold.

Figures 6, 7 and 8 compare the collision probability γ, the throughput per node θ and the mean packet delay $\bar{W}$ for a buffer size of $K = 5$. Similar results were obtained for K as large as 50 but not reported here due to space constraints. It can be seen that the SDAR approximation in *ns-2* and our finite buffer analysis both match extremely well with the unmodified *ns-2*. Furthermore, the results from SDAR approximation in *ns-2* and that from our finite buffer analysis are indistinguishable. This validates Approximation 4.1 and also establishes the fact that the observed small mismatches are due to the β_n's provided by the Bianchi-type model of [5]. In summary, the results in Figures 3-8 confirm that: 1) the SDAR model of contention can replace the CSMA/CA protocol to improve the simulation speed as described in Section VI without affecting the accuracy of results, and 2) the Markovian approximation of the coupled queue system is extremely accurate in predicting performance measures for Poisson traffic. Finally, we remark that our results are as accurate as that in [11]. However, we do not reproduce the results of [11] here since both their's and our's match extremely well with *ns-2* simulations.

VIII. Conclusion

In this paper, we applied the SDAR approximation to model the attempt process of nodes. We replaced the CSMA/CA mechanism by a coupled queue system with SDAR service. We developed an exact Markov model for the coupled queue system. We developed a technique to reduce the state space for the SDAR model and analyzed the finite buffer size case to be able to predict important performance measures. Under the Poisson traffic assumption, our simple model was shown to provide accurate results. We showed that the complexity of our model is less than an earlier model with same level of accuracy which happens to be the most accurate till date in terms of predicting the performance measures. We also reported how the SDAR approximation could be applied to improve the speed of *ns-2* simulations. Our work motivates to study the important theoretical question as to why the SDAR heuristic works well for the IEEE 802.11 networks. Modeling of non-homogeneous nodes with unequal Poisson arrival rates and with general traffic arrival processes such as the Batched Markovian Arrival Process (BMAP) are left as future work. We believe that this is feasible within the rigorous and formal framework laid out by this paper.

REFERENCES

[1] "Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications, ANSI/IEEE Std 802.11, 1999 Edition," vol. IEEE Std 802.11, 1999.

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

[3] C. Bordenave, D. McDonald, and A. Proutiere, "Random Multi-access Algorithms: A Mean Field Analysis," in *Allerton Conference on Communication, Control and Computing*, 2005.

[4] F. Cali, M. Conti, and E. Gregori, "Dynamic Tuning of the IEEE 802.11 Protocol to Achieve a Theoretical Throughput Limit," *IEEE/ACM Transactions on Networking*, vol. 8, no. 6, pp. 785–799, December 2000.

[5] A. Kumar, E. Altman, D. Miorandi, and M. Goyal, "New insights from a fixed point analysis of single cell IEEE 802.11 WLANs," *IEEE/ACM Transactions on Networking*, vol. 15, no. 3, pp. 588–601, June 2007, also appeared in INFOCOM 2005.

[6] F. Alizadeh-Shabdiz and S. Subramaniam, "Analytical Models for Single-Hop and Multi-Hop Networks," *Mobile Networks and Applications*, no. 11, pp. 75–90, 2006, springer Science.

[7] G. R. Cantieni, Q. Ni, C. Barakat, and T. Turletti, "Performance Analysis under Finite Load and Improvements for Multirate 802.11b," *Elsevier Computer Communication Journal*, April 2005.

[8] D. Malone, K. Duffy, and D. Leith, "Modeling the 802.11 Distributed Coordination Function in Non-saturated Heterogeneous Conditions," *IEEE/ACM Transactions on Networking*, vol. 15, no. 1, pp. 159–172, February 2007.

[9] O. Tickoo and B. Sikdar, "A Queueing Model for Finite Load IEEE 802.11 Random Access MAC," in *ICC*, 2004.

[10] K. Duffy and A. J. Ganesh, "Modeling the impact of buffering on 802.11," *IEEE Communication Letters*, vol. 11, no. 2, pp. 219–221, 2007.

[11] M. Garetto and C.-F. Chiasserini, "Performance Analysis of the 802.11 Distributed Coordination Function under Sporadic Traffic," in *Networking 2005*, Waterloo, Ontario, Canada, May 2005.

[12] R. Bruno, M. Conti, and E. Gregori, "Performance Modeling and Measurements of TCP Transfer Throughput in 802.11-based WLANs," in *MSWiM'06*, Torremolinos, Malaga, Spain, October 2-6, 2006 2006.

[13] G. Kuriakose, S. Harsha, A. Kumar, and V. Sharma, "Analytical Models for Capacity Estimation of IEEE 802.11 WLANs using DCF for Internet Application," *Wireless Networks*, 2007, to appear.

[14] E. D. Sykas, D. E. Karvelas, and E. N. Protonotarios, "Queueing Analysis of Some Buffered Random Multiple Access Schemes," *IEEE Transactions on Communications*, vol. 34, no. 8, pp. 790–798, 1986.

[15] H. Kobayashi, *Modeling and Analysis*. Addison Wesley, 1978, pp. 201–205.

[16] G. Fayolle, V. Malyshev, and M. Menshikov, *Topics in the Constructive Theory of Markov Chains*. Cambridge University Press, 1995.

APPENDIX

We use Foster's criterion (see [16]) to prove Theorem 3.1.

Theorem A.1: A time homogeneous irreducible aperiodic Markov chain $\{X(t), t \geq 0\}$ with a countable state space $\mathcal{S}$ is positive recurrent if and only if there exists a non-negative function $f(x)$, $x \in \mathcal{S}$, a number $\epsilon > 0$, and a finite set $A \subset \mathcal{S}$, such that the following conditions hold:

1) $\mathsf{E}\Big(f(X(t+1)) - f(X(t))\Big| X(t) = x\Big) \leq -\epsilon, \forall x \in \mathcal{S} \setminus A$

2) $\mathsf{E}\Big(f(X(t+1))\Big| X(t) = x\Big) < \infty, \forall x \in A.$

Clearly, the DTMC $\{\boldsymbol{Q}(t), t \geq 0\}$ is time homogeneous and has a countable state space. Irreducibility and aperiodicity can be easily proved. Any state $(k_1, k_2, \cdots, k_M) \in \mathbb{N}^M$ can be reached from the state $(0, 0, \cdots, 0)$ in one step by k_i, $1 \leq i \leq M$, arrivals to the i^{th} queue. Similarly, the state $(0, 0, \cdots, 0)$ can be reached from any state $(k_1, k_2, \cdots, k_M) \in \mathbb{N}^M$ in $\sum_{i=1}^M k_i$ steps by $\sum_{i=1}^M k_i$ consecutive successes and no arrivals such that the k_i's do not increase in between. Hence, $\{\boldsymbol{Q}(t), t \geq 0\}$ is irreducible. Since $\{\boldsymbol{Q}(t), t \geq 0\}$ is irreducible and there exists a self loop, e.g., from the state $(0, 0, \cdots, 0)$ to itself, the DTMC $\{\boldsymbol{Q}(t), t \geq 0\}$ is aperiodic as well.

We define the finite set A and the non-negative function $f(\cdot)$ as follows:

$$A := \{\mathbf{0}\} = \{(0, 0, \cdots, 0)\} \; ; \; f(k_1, k_2, \cdots, k_M) := \sum_{i=1}^M k_i$$

Notice that $A := \{\mathbf{0}\} = \{(0, 0, \cdots, 0)\}$ corresponds to the system empty state, and for given $k_1, k_2, \cdots, k_M$, the function $f(k_1, k_2, \cdots, k_M) := \sum_{i=1}^M k_i$ gives the total number of packets in the system. It can be shown that

$$\mathsf{E}\Big(f(\boldsymbol{Q}(t+1))\Big| \boldsymbol{Q}(t) = \mathbf{0}\Big) = M\lambda\sigma \tag{17}$$

which is finite as long as the arrival rate λ is finite. Hence, for the DTMC $\{\boldsymbol{Q}(t), t \geq 0\}$, the second condition of Theorem A.1 holds for all finite arrival rates. Note that when the system is in the state $\mathbf{0} = (0, 0, \cdots, 0)$, only an idle channel slot can occur and $M\lambda\sigma$ represents the mean number of arrivals to the system in an idle channel slot of duration σ. It can also be shown that, $\forall \boldsymbol{k} = (k_1, k_2, \cdots, k_M)$ such that $\sum_{i=1}^M \mathbf{1}_{\{k_i > 0\}} = n$, $1 \leq n \leq M$, we have

$$\begin{aligned} & \mathsf{E}\Big(f(\boldsymbol{Q}(t+1)) - f(\boldsymbol{Q}(t))\Big| \boldsymbol{Q}(t) = \boldsymbol{k}\Big) \\ = \; & M\lambda\sigma\left(1 + p_{succ,n}T_s + p_{coll,n}T_c\right) - p_{succ,n} \end{aligned} \tag{18}$$

Since ϵ can be made arbitrarily small we observe that for the DTMC $\{\boldsymbol{Q}(t), t \geq 0\}$, the first condition of Theorem A.1 holds if

$$M\lambda < \frac{p_{succ,n}}{\sigma\left(1 + p_{succ,n}T_s + p_{coll,n}T_c\right)}$$

for all $1 \leq n \leq M$. Notice that

$$L_{sat,n} := \sigma\left(1 + p_{succ,n}T_s + p_{coll,n}T_c\right),$$

$$\text{and} \quad \Theta_{sat,n} := \frac{p_{succ,n}}{L_{sat,n}}$$

are, respectively, the mean channel slot duration and the mean aggregate throughput (in packets/sec) in a WLAN consisting of n saturated nodes. Hence, for the DTMC $\{\boldsymbol{Q}(t), t \geq 0\}$, the first condition of Theorem A.1 holds if

$$M\lambda < \Theta_{sat,n}$$

for all $1 \leq n \leq M$, i.e., if $M\lambda < \min_{1 \leq n \leq M} \Theta_{sat,n}$. Since the second condition of Theorem A.1 holds at all finite arrival rates, the DTMC $\{\boldsymbol{Q}(t), t \geq 0\}$ is positive recurrent if $M\lambda < \min_{1 \leq n \leq M} \Theta_{sat,n}$.