An Exact Solution to An Approximated Model of RED

Claus Bauer
Dolby Laboratories
San Francisco, CA, 94103, USA
Email: cb@dolby.com

Homayoun Yousefi'zadeh Hamid Jafarkhani
Department of EECS
University of California, Irvine, USA
Email: {hyousefi,hamidj}@uci.edu

***Abstract*— In this paper, we propose an analytical model to capture the dynamics of the RED algorithm. We first develop a system of recursive equations that describes the packet dropping behavior of the RED algorithm. Using a notion from the theory of random walks, we then derive an exact-closed form expression that characterizes the loss characteristics of a RED queue. We validate the derived formula by a numerical comparison with the recursive equations.**

I. Introduction

The traditional Internet architecture relies on a best-effort architecture for all kinds of data traffic. The feedback-based TCP protocol has initially been used to provide reliable data services. However, the emergence of real-time applications puts strict latency demands on the network infrastructure that cannot be guaranteed by the TCP protocol. The IETF has introduced the integrated and the differentiated service technologies to provide stronger quality of service guarantees.

The quality of service performance of a network also depends on the packet drop mechanisms of routers. The simplest dropping schemes drop all packets arriving at a full queue. Active Queue Management (AQM) has been proposed as a mean to alleviate some congestion problems. Variations of AQM include Early Packet Discard [7], and Random Early Drop [11]. A widely studied AQM scheme is Random Early Detection (RED) [4], [5]. Newer AQM schemes such as Adaptive Virtual Queue Algorithm [8], Random Exponential Marking [1] have been shown to address several shortcomings of RED. However, the RED algorithm continues to be used in practice and any new insights in its performance are of benefit to network operators.

From a practical stand point, the deployment of the RED algorithm is difficult because it must configured by choosing the values of several parameters. The optimal configuration of these parameters is not well understood.

Using simplified assumptions, guidelines for setting the RED parameters have been proposed in [5]. Most of these studies [3] are based on heuristics or simulations, but not on a strict mathematical analysis. In [2], the RED algorithm is modelled stochastically, whereas in [10] a Markov based approach is used to study the optimal tuning of the RED algorithm.

Most of the aforementioned research has focused on tuning the RED parameters, namely, the maximum and minimum thresholds. The average loss rate of a RED queue has not been studied in detail. A model of the loss behavior of the RED algorithm was first introduced in [13]. The authors model the RED queue as a 3 state Markov model. An approximated closed-form solution of the Markov model is derived. While the solution provides satisfactory approximated results, it still different from the precise values.

In this paper, we improve on the work of [13] by deriving an exact closed-form solution for a 4-state Markov model of a RED queue. The Markov model is derived by approximating an exact queueing model of a RED queue. Introducing a notion from the theory of random walks, we then derive an exact solution of the Markovian RED model. In particular, our model allows us to derive the expected loss probabilities over arbitrary long periods of time. We calculate the average probability of dropping k out of n packets which allows the design of optimal Reed-Solomon codes.

In Section II, we first define an exact queueing model of a RED queue which we then approximate by a Markov model of the RED queue. We present a closed form solution of the Markov model in Section III. We evaluate our results numerically in Section IV and conclude in Section V.

II. Model and terminology

We first define an exact queueing system of the RED algorithm. For the subsequent analysis of the RED algorithm, we then convert the queueing system into an 4-state Markov model which approximately describes the dynamics of RED and which we solve exactly in the following section. Finally, we present a recursive formula that describes the loss behavior of the RED algorithm.

A. RED algorithm

The RED algorithm is typically applied to a queue at the output of a switch. A packet arriving at the queue is dropped with probability p calculated based on the average queue size q_t. The average queue size q_t is calculated using an exponential weighted moving average as

$$q_t = (1 - w_q)q_{t-1} + w_q\tilde{q}_t.$$

Here, q_t is the current average queue size, q_{t-1} is the average queue size at the last time instant, w_q is the weighting function defining the exact low-pass filter, and $\tilde{q}_t$ is the current instantaneous queue size. q_t is then compared to two thresholds, a minimum threshold q_{min} and a maximum threshold q_{max}. Each arriving packet is dropped with probability p given by

$$p = \left\{ \begin{array}{ll} 0, & q_t < q_{min} \\ \frac{q_t - q_{min}}{q_{max} - q_{min}} p_{max}, & q_{min} \le q_t \le q_{max}, \\ 1, & q_{max} < q_t, \end{array} \right\} \quad (1)$$

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE GLOBECOM 2007 proceedings.

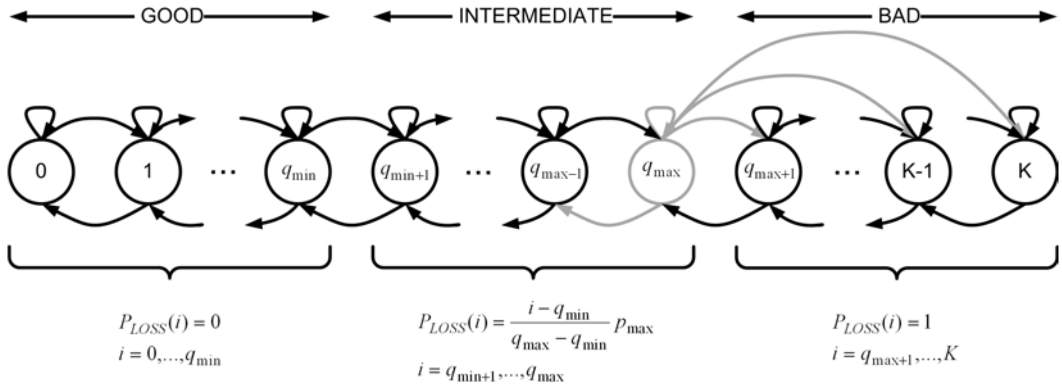

Fig. 1. M/D/1/K approximation of the steady-state behavior of RED under quasi-stationary assumptions. For clarity, only the full set of transitions associated with $k = q_{max}$ are shown.

where $p_{max} \leq 1$.

B. Exact queueing model of RED queue

We consider a queuing system with the capacity of K fixed size packets operating under the RED queuing discipline. A fixed packet size can represent an atomic unit of operation when receiving variable length packets and thus does not represent a loss of generality. We define the steady states of the queue occupancy model as $U_0, .., U_K$. We denote the transition probabilities as u_{ij}, $0 \leq i, j \leq K$, where some $u_{ij} = 0$.

The RED queuing system is described by its traffic pattern and is assumed to be operating in its steady-state regime. In [10], the authors develop a model for analyzing both transient and steady-state behavior of RED queues accommodating a large population of random traffic sources the traffic generation pattern of which is described by a Poisson arrival process with a time varying rate. As the result of enforcing RED packet discarding mechanism with a small averaging factor of w_q in the order of 10^{-3} and for a slowly varying Poisson parameter, the RED queue is considered to be operating in a quasi-stationary state. As such, the behavior of the queue can be approximated with M/G/1/K queuing discipline. For the purpose of our study, we select the M/D/1/K queuing discipline, as shown in Fig. 1, not only as a special case of M/G/1/K but as the best practical alternative. For an M/D/1/K queue with a load factor ρ, we normalize the service time to indicate the time unit such that the arrival intensity is equal to ρ. Then, the steady-state probabilities π_k of being in state k for $k \in \{1, \cdots, K\}$ form a discrete Probability Density Function the terms of which are calculated as

$$\pi_k = \begin{cases} \frac{\pi_k^\infty}{\pi_0^\infty + \rho G(K)}, & \text{if} \quad k \in \{0, \cdots, K-1\} \\ 1 - \frac{G(K)}{\pi_0^\infty + \rho G(K)}, & \text{if} \quad k = K \end{cases} \tag{2}$$

where $G(K) = \sum_{k=0}^{K-1} \pi_k^\infty$. Further, the steady-state probability π_k^∞ of state k for an infinite capacity M/D/1 queuing system with load ρ is identified in Page 44 of [6] as

$$\begin{aligned} \pi_k^\infty \;&= (1-\rho)\,[\,\textstyle\sum_{i=1}^{k} e^{\rho i}(-1)^{k-i}\frac{(i\rho)^{k-i}}{(k-i)!} \\ &\quad + \textstyle\sum_{i=1}^{k-1} e^{\rho i}(-1)^{k-i}\frac{(i\rho)^{k-i-1}}{(k-i-1)!}\,], \quad k \geq 2 \end{aligned} \tag{3}$$

with $\pi_0^\infty = 1 - \rho$ and $\pi_1^\infty = (1-\rho)(e^\rho - 1)$.

In this paper, we assume the average load ρ to be constant. In practice, data traffic is often sent using the TCP protocol which adjusts its sending rate in reaction to the dropping behavior of the RED queue. We are currently investigating how our approach could be extended to this scenario.

C. 4-state Markov approximation model

We now use the exact queueing model from the previous section to derive an Markov approximation model for the RED algorithm. We use the model to derive a closed form solution for the average probability of dropping k out of n packets.

According to the way p is defined in eqn. (1), we can define three *aggregate* states that characterize the behavior of the RED algorithm. We define the bad aggregate state b, the intermediate aggregate state I, and the good aggregate state g, where packets are dropped with probability $p = 1$, $p = \frac{q_t - q_{min}}{q_{max} - q_{min}} p_{max}$, and $p = 0$, respectively. We divide the integers in the interval $[0, K]$ into three subsets $S_g := \{1, .., q_{min}\}$, $S_I := \{q_{min+1}, .., q_{max}\}$, $S_b = \{q_{max+1}, .., q_K\}$. From Fig. 1, we see that $b = \bigcup_{i \in S_b} U_i$, $I = \bigcup_{i \in S_I} U_i$, $g = \bigcup_{i \in S_g} U_i$.

We further divide the aggregate state I in two sub-aggregate states I^b and I^g where I^b denotes the sub-aggregate state of I where a packet is dropped. Similarly, I^g denotes the sub-aggregate state of I where a packet is *not* dropped. For further usage, we set $S_{I_b} = S_{I_g} = S_I$. For p defined as in eqn. (1) and assuming the RED queue is in aggregate state I, I^b occurs with conditional probability p and I^g occurs with conditional probability $1-p$. We note that whereas for the aggregate state I the loss probability p of an arriving packet is a random variable, for each of the sub-aggregate states I^b and I^g the loss behavior is deterministic, i.e., $p = 0$ and $p = 1$, respectively.

We now derive the transition probabilities p_{ij}, $i, j \in \{b, I^b, I^g, g\}$, where p_{ij} denotes the probability that the RED queue changes from aggregate state i to aggregate state j upon the arrival of a packet, from the queueing model given in Fig. 1. The transition probability p_{ij} is the probability that the system changes from any state U_k in the set S_i to any state U_l in the set in the change S_j. By the definition of the states I_b (and I_g), we see that the probability that state U_i, $i \in S_I$, belongs to I_b is $P_{LOSS}(i)$. For I_g, it is $1 - P_{LOSS}(i)$. Setting

$$\pi_i^* \;=\; \left\{ \begin{array}{ll} \pi_i, & i \in \{S_b, S_g\}, \\ \pi_i P_{LOSS}(i), & i \in S_{I_b}, \\ \pi_i(1 - P_{LOSS}(i)), & i \in S_{I_g}, \end{array} \right\} \quad (4)$$

$$u_{ij}^* \;=\; \left\{ \begin{array}{ll} u_{ij}, & j \in \{S_b, S_g\}, \\ u_{ij} P_{LOSS}(j), & j \in S_{I_b}, \\ u_{ij}(1 - P_{LOSS}(j)), & j \in S_{I_g}, \end{array} \right\} \quad (5)$$

we derive

$$p_{ij} = \frac{\sum\limits_{k \in S_i} \sum\limits_{l \in S_j} \pi_k^* u_{kl}^*}{\sum\limits_{k \in S_i} \pi_k^*}, \quad i, j \in \{b, I^b, I^g, g\}. \quad (6)$$

We see from (4) and (6) that the transition probabilities p_{ij} are a function of the loss probabilities $P_{LOSS}(i)$.

In the sequel of this paper, we will only refer to the four *aggregate states* b, I^b, I^g, g and refer to them simply as *states*.

D. Recursive definition of loss probabilities

We first calculate the quantities $P(v + z, v, j)$, $j \in \{b, I^b, I^g, g\}$ which are defined as the probabilities of dropping z out of $(v + z)$ arrived packets and being in state j after the $(v + z)$-th packet is buffered/dropped by the RED algorithm. Summing over the loss probabilities of the 4 different states of the RED queue, we obtain the overall probability $P(v + z, v)$ to lose z out of $(v + z)$ packets as

$$P(v + z, v) \;=\; \sum_{j \in \{b, I^b, I^g, g\}} P(v + z, v, j). \quad (7)$$

We now calculate the loss probabilities $P(v + z, v, j)$ as functions of the transition probabilities p_{ij} and the steady state probabilities j_{ss}, $j \in \{b, I^b, I^g, g\}$, where j_{ss} is the probability that the RED queue is in state j.

We describe the dynamics of a RED queue via the following

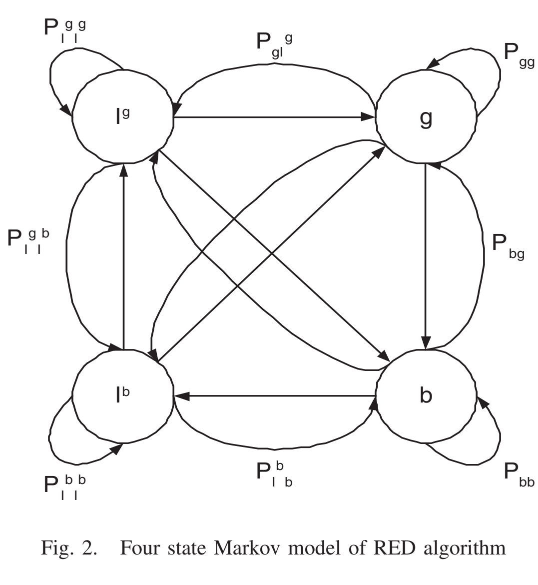

Fig. 2. Four state Markov model of RED algorithm

recursive relations. For $c \in \{g, I_g\}$,

$$P(v + z, v, c) = \sum_{d \in \{b, I^b, I^g, g\}} P(v - 1 + z, v - 1, d) p_{dc}. \quad (8)$$

For $c \in \{b, I_b\}$,

$$P(v + z, v, c) = \sum_{d \in \{b, I^b, I^g, g\}} P(v + z - 1, v, d) p_{dc}. \quad (9)$$

The loss probabilities for a single packet can be calculated from the transition and steady state probabilities as follows:

$$\begin{array}{rcll} P(1, 0, j) & = & 0, \; j \in \{g, I^g\} & (10) \\ P(1, 0, j) & = & p_{gj} g_{ss} + p_{I^g j} I^g_{ss} + p_{bj} b_{ss} + p_{I^b j} I^b_{ss} & (11) \\ & & j \in \{b, I^b\}, & \\ P(1, 1, j) & = & 0, \; j \in \{b, I^b\} & (12) \\ P(1, 1, j) & = & p_{gj} g_{ss} + p_{I^g j} I^g_{ss} + p_{bj} b_{ss} + p_{I^b j} I^b_{ss} & (13) \\ & & j \in \{g, I^g\}. & \end{array}$$

Similar to eqn. (8) - (9), one can formulate a system of equations to calculate the initial steady state probabilities: For $c \in \{b, I^b, I^g, g\}$,

$$c_{ss} \;=\; \sum_{d \in \{b, I^b, I^g, g\}} d_{ss} p_{dc}. \quad (14)$$

The system of equations (14) can be solved by noting that only three of the four equations are linearly independent and that the sum of the four steady state probabilities equals one.

We introduce some further terminology. In order to express the quantities $P(v + z, v, j)$ as a function of the steady state probabilities j_{ss}, we have to consider all possible scenarios which lead to z drops out of $(v + z)$ arrived packets and leave the RED queue in to state j after the $(v + z)$-th packet has been buffered/dropped by the RED algorithm. In other words, we have to look at all random walks of length $(v + z + 1)$ that start in an initial steady state j_{ss} with $j \in \{b, I^b, I^g, g\}$, that after the buffering/dropping of the $t = v + z$-th packet finishes in state j, and that after the buffering/dropping after any of the intermediate packets $t = 1, .., v + z - 1$ are in one of the four states $j \in \{b, I^b, I^g, g\}$. For the calculation of the packet loss probabilities, we assume that no packet is either received or gets lost in the initial state j_{ss}. The transition probabilities between the different states at consecutive steps are the transition probabilities $p_{ij}, 1 \leq i, j \leq 4$ defined above.

We further define a 4×4 integer matrix $a = (a_{ij})_{i,j \in \{b, I^b, I^g, g\}}$. We define the functions $f^{i,j}(n{+}1, a)$ as the number of possibilities to construct a random walk of length $n{+}1$ that starts in state i, finishes in state j, and that for each $i, j \in \{b, I^b, I^g, g\}$ contains $0 \leq a_{i,j} \leq n$ transitions from state i to j. We require of course that $\sum\limits_{i,j \in \{b, I^b, I^g, g\}}^{4} a_{i,j} = n$. We note that the function $f^{i,j}(n + 1, a)$ can be equal to zero. For example, if $n = 2$, $i = j = I^g$, $a_{g,g} = 2$, and all other $a_{i,j} = 0$, then $f^{I^g, I^g}(a) = 0$ as it is not possible to construct a random walk of length n that starts and ends at I^g, when only $a_{ij} = 0$ with $(i, j) \neq (g, g)$.

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE GLOBECOM 2007 proceedings.

III. Solution of the RED Markov model

In eqn. (8) - (9) we have defined a set of recursive equations that allow us to calculate the quantities $P(v+z, v, j)$. In this section, we derive closed-form solutions for the $P(v+z, v, j)$. In particular, we show the following theorem:

Theorem 1: For $v+z \geq 1$, $c \in \{b, I^b, I^g, g\}$

$$P(v+z, v, c) = \sum_{d \in \{b, I^b, I^g, g\}} D^{v,z,d,c} d_{ss}. \tag{15}$$

where for $i, j \in \{b, I^b, I^g, g\}$:

$$\begin{aligned} & D^{v,z,i,j} \\ = & \sum_{a_{gg} \geq 0}^{v,z,*} \sum_{a_{gI^g} \geq 0}^{v,z,*} \sum_{a_{gb} \geq 0}^{v,z,*} \sum_{a_{gI^b} \geq 0}^{v,z,*} \cdots \sum_{a_{I^b b} \geq 0}^{v,z,*} \sum_{a_{I^b I^b} \geq 0}^{v,z,*} \\ & f^{i,j}(v+z+1, a) p_{gg}^{a_{gg}} p_{gI^g}^{a_{gI^g}} p_{gb}^{a_{gb}} p_{gI^b}^{a_{gI^b}} .. p_{I^b b}^{a_{I^b b}} p_{I^b I^b}^{a_{I^b I^b}} . \end{aligned} \tag{16}$$

The notation $\sum_{a_{ij} \geq 0}^{v,z,*}$ denotes only summing over such a_{ij} that satisfy the following conditions:

$$\sum_{j \in \{b, I^b, I^g, g\}} a_{jb} + \sum_{j \in \{b, I^b, I^g, g\}} a_{jI^b} = z \tag{17}$$

$$\sum_{j \in \{b, I^b, I^g, g\}} a_{jg} + \sum_{j \in \{b, I^b, I^g, g\}} a_{jI^g} = v, \tag{18}$$

where the 4×4 matrix $a = (a_{ij})_{i,j \in \{b, I^b, I^g, g\}}$ is determined by the values a_{ij} over which the terms $D^{v,z,i,j}$ are summed. **Remark:** Closed-form expressions for the functions $f^{i,j}(n, a)$ are not known. However, the actual values of $f^{i,j}(n, a)$ do not depend on the transition and steady state probabilities. Thus, they can be calculated off-line by an exhaustive search.

A. *Proof of Theorem 1*

In the definitions of the terms $P(v+z, v, j)$, the quantity $k := v+z$ expresses the number of packets that have been received at a RED queue during the observation period. We prove Theorem 1 by induction over k. This means that for the inductive step, we assume that the eqn. (15) have been shown for all pairs v and z that satisfy $k = v+z$, and under this assumption show that they also hold for all pairs v and z satisfying $k+1 = v+z$.

Inductive base case k=1: We start with the case $v = 0$, $z = 1$. We first consider the term $P(1, 0, g)$. We have to show that for $j = g$ eqn. (10) follows from eqn. (15) with $c = g$. We see from eqn. (15) with $c = g$ that it is sufficient to show that $D^{0,1,j,g} = 0$ for $j \in \{b, I^b, I^g, g\}$. Eqn. (18) with $v = 0$ implies that $a_{jg} = 0$ for $j \in \{b, I^b, I^g, g\}$. However, $a_{jg} = 0$, $j \in \{b, I^b, I^g, g\}$ implies that $f^{j,g}(2, a) = 0$ as there is no possibility to construct a random walk of length 2 that ends at g if there are no transitions from any state $j \in \{b, I^b, I^g, g\}$ to state g. Thus, $D^{0,1,j,g} = 0$. The same argument shows that $P(1, 0, I^g)$ is defined correctly by eqn. (15) with $c = I_g$.

We now consider the term $P(1, 0, b)$. We show that eqn. (11) follows from eqn. (15) for $c = b$. We see from (18) with $v = 0$ that for each $D^{0,1,j,b}$ there is $a_{jg} = a_{jI^g} = 0$ for $j \in \{b, I^b, I^g, g\}$. Thus, we are only left with terms a_{jI^b} and a_{jb}. We note that for a random walk of length 2, for each j exactly one of the entries a_{jb} or a_{jI^b} is not equal to zero. If this entry is of the type a_{jI^b}, then $f^{j,b}(2, a) = 0$ because a random walk of length 2 cannot terminate at b if only transitions of type a_{kI^b} happen. If the only non-zero entry of matrix a is of type a_{jb}, there is exactly one possibility of constructing a random walk of length 2 that begins at j and ends at b, i.e., $f^{j,b}(2, a) = 1$. Summarizing the findings of this paragraph, we see that eqn. (11) follows from eqn. (15) with $c = b$. We can show in the same way that the quantity $P(1, 0, I^b)$ is defined correctly by eqn. (15) with $c = I_b$.

Second, we consider the case $v = 1$ and $z = 0$. We show that for $j = b$ (and $j = I^b$) eqn. (12) follows from eqn. (15) with $c = b$ ($c = I_b$). We argue similarly as above to show that $f^{j,b}(2, a)$ and $f^{j,I^b}(2, a)$ are equal to zero. In particular, we show that eqn. (17) with $z = 0$ implies $f^{j,b}(2, a) = 0$ (and $f^{j,I^b}(2, a) = 0$). Next, we consider the term $P(1, 1, g)$. We have to show that eqn. (13) follows from eqn. (15) with $c = g$.. We see from eqn. (17) for $z = 0$ that all $a_{jb} = a_{jI^b} = 0$. Arguing as above, we see that for each j exactly one of the two entries a_{jg} or a_{jI^g} is not equal to zero. We further follow the argument above to show that in the first case $f^{j,g}(2, a) = 1$ and in the second case $f^{j,g}(2, a) = 0$,. The expression $P(1, 1, I^g)$ is treated in the same way.

Inductive step from k to k+1: We now show that if the eqns. (15) are true for all pairs v and z satisfying $k = v+z$, then they are also true for all pairs v and z satisfying $k+1 = v+z$. We first consider the quantity $P(v+z, v, g)$. By the inductive assumption, we see that on the right hand side of eqn. (8) with $c = g$, we have $k = v-1+z$. Thus, we can apply the inductive assumption to the right hand side of eqn. (8) with $c = g$. We insert the eqn. (15) in both sides of eqn. (8) with $c = g$. We have to show that for $c \in \{b, I^b, I^g, g\}$, there is

$$D^{v,z,c,g} = \sum_{d \in \{b, I^b, I^g, g\}} D^{v-1,z,c,d} p_{dg}. \tag{19}$$

In order to show eqn. (19) with $c = g$, we proceed in 2 steps:

1) $D_{v,z,i,j}$ is defined in eqn. (16) via a multiple summation over the terms a_{ij}. Each of the specific sixteen tuples $a_{i,j}$, $i, j \in \{b, I^b, I^g, g\}$ occurring in the multiple summation defines a matrix a. We show that a matrix a appears on the left hand side of eqn. (19) with $c = g$, i.e., in the summation defining $D^{v,z,g,g}$, if and only if it also appears on the right hand side of eqn. (19) with $c = g$, i.e., for at least one of the terms $D^{v-1,z,g,g} p_{gg}$, $D^{v-1,z,g,I^g} p_{I^g g}$, $D^{v-1,z,g,b} p_{bg}$, or $D^{v-1,z,g,I^b} p_{I^b g}$ it is part of the summation defining the term.
2) For a fixed matrix a appearing on both sides of eqn. (19) with $c = g$, we show that the sum of the terms $f^{i,j}(\cdot, \cdot)$ which are counted when a appears in the summation def. a term D or Dp is equal on both sides of the eqn.

The relation (19) with $c = g$ then follows from both claims.

In order to show the first claim, we distinguish two cases. In the first case $D^{v,z,g,g} = 0$. The definition (16) of $D^{v,z,g,g}$

implies that $f^{g,g}(\cdot,\cdot) = 0$ for all terms $f^{g,g}(\cdot,\cdot)$ appearing in the definition of $D^{v,z,g,g}$. In other words, there exists no possible random walk of length $(v+z+1)$ starting and terminating in state g. This implies that there exists no random walk of length $(v+z)$ starting in state g and terminating in any state j. Otherwise, one could construct a random walk of length $(v+z+1)$ starting and terminating in g by taking a random walk of length $(v+z)$ that starts in g and terminates in j and adding an additional transition from j to g. Thus, the terms $f^{\cdot,\cdot}(\cdot,\cdot)$ appearing in the definitions of the terms D on the right hand side are all equal to zero. This proves eqn. (19) with $c = g$ when $D^{v,z,g,g} = 0$.

In the second case, in the definition of $D^{v,z,g,g}$ we sum over at least one matrix a such that $f^{g,g}(v+z+1,a) \neq 0$. For such a matrix a, the definition of $f^{g,g}(v+z+1,a)$ implies that there are u, $1 \leq u \leq 4$ states j_i, with $1 \leq i \leq u$, $j_i \in \{b, I^b, I^g, g\}$ such that $a_{j_i g} \neq 0$.

We now define a 4×4 matrix J_{ij} as the matrix with the (i,j) entry equal to 1 and all other entries equal to 0. We define $a - J_{ij}$ as the matrix obtained by the per-entry subtraction of the matrix entries. For any matrix a with $f^{g,g}(v+z+1,a) \neq 0$ and j_i such that $a_{j_i g} \neq 0$, one can easily verify that the matrix $a - J_{j_i g}$ appears in the summation (16) defining D^{v-1,z,g,j_i}. Multiplying D^{v-1,z,g,j_i} by $p_{j_i g}$ on the right hand side of eqn. (19) with $c = g$ changes the matrix $a - J_{j_i g}$ to a. In consequence, each matrix a appearing on the left-hand side of eqn. (19) with $c = g$ also appears on the right-hand side for the terms D^{v-1,z,g,j_i} when $a_{j_i g} \neq 0$. The inverse statement follows in the same way. Thus, the second claim follows from Lemma 1 the proof of which we omit for space reasons:

Lemma 1: For $k \geq 1$,

$$f^{i,j}(k+1,a) = \sum_{l \in \{b, I^b, I^g, g\},\, a_{l,j} > 0} f^{i,l}(k, a - J_{l,j}).$$

For $c \neq g$, the eqn. (19) is shown in the same way as for $c = g$. The inductive steps for eqn. (15) with $c \neq g$ are shown in the same way as for $c = g$. $\square$

IV. Numerical results

In this section, we numerically evaluate the performance of the RED algorithm using Theorem 1. As an input for a numerical evaluation, we require values for the transition probabilities p_{ij} as defined in sec. II. In [13], a methodology to derive these values from actual simulations has been described. We have used the methodology proposed in [13] and have derived the transition probabilities given in Table I. Using these values, we calculate the values of $P(v+z,v)$ as defined in eqn. (7). The results are shown in Fig.3.

We have calculated the values $P(v+z,v)$ using eqn. (7). The values $P(v+z,v,j)$ can be obtained using either the recursive eqn. (8) - (9) or via the closed-form expressions of Theorem 1. As Theorem 1 provides an exact solution for the quantities $P(v+z,v,j)$, we expect that both ways to calculate $P(v+z,v,j)$ give identical results. Our numerical experiments confirm this expectation. This improves over the previous work in [13] where one could observe numerical differences between the recursive and the closed-form solutions.

TABLE I

Transition probabilities

	b	I^b	g	I^g
b	0.6	0.3	0.0	0.1
I^b	0.3	0.4	0.1	0.2
g	0	0.1	0.36	0.54
I^g	0.1	0.2	0.2	0.5

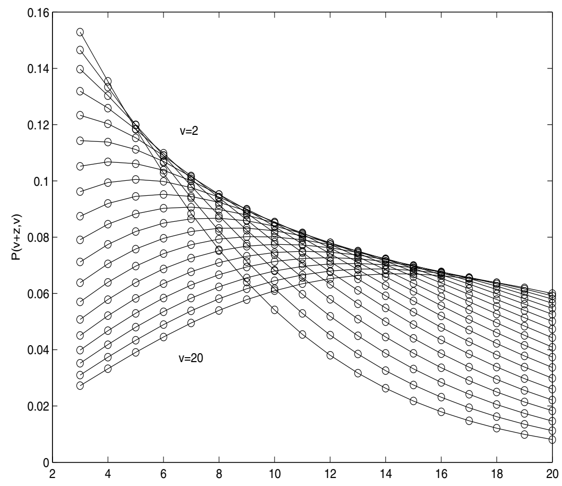

Fig. 3. $P(v+z,v)$ for $v \in [2,20]$

V. Conclusions

In this paper, we develop an exact queueing model of the RED algorithm which we convert into a four state Markov model. We use the Markov model to describe the loss characteristics of a RED queue. We derive an exact closed-form solution of the Markov model. We show via simulations that the closed-form solution matches exactly with a recursive description of the loss process.

References

[1] S. Athuraliya, V. H. Li, S.H. Low, Qinghe Yin, " REM: Active Queue Management," IEEE Network, May 2001.

[2] H.M. Alazemi, A. Mokhtar, M. Azizoglu, "Stochastic Approach for Modeling Random Early Detection Gateways in TCP/IP Networks," In Peoc. IEEE ICC, 2001.

[3] T. Bonald, M. May, J. Bolot, "Analytic Evaluation of RED Performance," In Proc. IEEE INFOCOM, 2000.

[4] R. Braden et al., "Recommendations on Queue Management and Congestion Avoidance in the Internet," IETF RFC 2309.

[5] S. Floyd, V. Jacobson, "Random Early Detection Gateways for Congestion Avoidance," IEEE/ACM Trans. Networking, August 1993.

[6] E. Gelenbe, G. Pujolle, "Introduction to Queueing Networks," John Wiley & Sons, Inc., ISBN 0-471-96294-5, 1998.

[7] V. Jacobson, "Presentations to the IETF Performance and Congestion Control Working Group," August 1989.

[8] S. Kunniyur and R. Srikant, "Analysis and Design of An Adaptive Virtual Queue (AVQ) Algorithm for Active Queue Management, In Proc. ACM SIGCOMM, 2001.

[9] P. Lassila, J. Virtamo, "Modeling the Dynamics of the RED Algorithm," In Proc. ACM Quality of Future Internet Services (QofIS), 2000.

[10] P. Kuusela, P. Lassila, J. Virtamo, P. Key, "Modeling RED with Idealized TCP Sources," In Proc. IFIP Performance Modeling and Evaluation of ATM & IP Networks, 2001.

[11] A. Romanov, S. Floyd, "Dynamics of TCP Traffic over ATM networks," In Proc. ACM SIGGCOMM, 1994.

[12] H. Sanneck, G. Carli, R. Koodli, "A Framework Model for Packet Loss Metrics Based on Loss Runlengths," In Proc. SPIE/ACM SIGMM Mulitmedia Computing and Networking, 2000.

[13] M.B. Singh, H. Yousefi'zadeh, H. Jafarkhani, "A Finite State Markov Chain Model for Statistical Loss across A RED Queue," In Proc. IEEE/IARIA ICHSN, 2005.