Hybrid Random Early Detection Algorithm for Improving End-to-End Congestion Control in TCP/IP Networks

Aun Haider, Harsha Sirisena and Krzysztof Pawlikowski

Technical Report TR-COSC 02/05

University of Canterbury,

Christchurch,

New Zealand

E-mails: {aha38@it, sirisehr@elec, krys@cosc}.canterbury.ac.nz

April 3, 2005

Abstract

The successful operation of the present Internet depends mainly upon TCP/IP which employs end-to-end congestion control mechanisms built in the end hosts. In order to further enhance this paradigm of end-to-end control the Random Early Detection algorithm has been proposed, which starts to mark or drop packets at the onset of congestion. The paper addresses issues related to the choice of queue length indication parameter for packet marking/dropping decisions in RED-type algorithms under the varying traffic conditions. Two modifications to RED are proposed: (i) use of both instantaneous queue size and its EWMA for packet marking/dropping and (ii) reducing the effect of the EWMA queue size value when the queue size is less then min_{th} for a certain number of consecutive packet arrivals.

The newly developed Hybrid RED algorithm can effectively improve the performance of TCP/IP based networks while working in a control loop formed by either dropping or marking of packets at congestion epochs. New guidelines are developed for better marking/dropping of packets to achieve a faster response of RED-type algorithms. The hybrid RED algorithm has been tested using ns-2 simulations, that show better utilization of network bandwidth and a lower packet loss rate.

Contents

Chapter 1

Introduction

TCP is a reliable end-to-end transport protocol, widely used in the current Internet for such diverse applications as email, Usenet news, remote login, file transfer, some streaming audio and video protocols and www. As a result of this wide spread proliferation and integration of TCP/IP code in commonly used operating systems, such as Unix, Linux and Microsoft Windows, it constitutes a major portion of the current Internet traffic.

The growth of networks operating under TCP/IP has been unprecedented. The critical factor in the robustness of the current best effort Internet turns out to be end-to-end congestion control mechanisms. In this mechanism TCP constantly adapts itself to available bandwidth slowing down the data transfer rate when it detects congestion and speeding up again when there is no congestion. Traditionally these end-to-end congestion control mechanisms were incorporated only in the end host so as to aid in upgrading the software and keeping the functions of core routers simpler. However, due to the present mammoth size of the current Internet and the unreliability of developers it has been suggested to involve routers in performing the vital task of congestion control in the Internet; see e.g. [Braden et al. 1998] and [Floyd and Fall 1999].

Presently, most widely used routers employ a Droptail type policy which has the inherent limitation of inability to convey congestion information to the end host. Furthermore they have other well known problems such as flow synchronization; see Floyd and Jacobson [1993] and [Floyd and Jacobson 1994].Thus, an alternative mechanism is desired, at the congestion prone routers, to detect the onset of congestion and accordingly send feedback to the end hosts, forcing reduction in their data sending rates. Thereby, congestion and consequent loss of packets is reduced. The congestion feedback from a router can be conveyed back

to end hosts by Explicit Congestion Notification (ECN), to form so-called Active Queue Management (AQM); see [Ramakrishnan et al. 2001]. Recently several algorithms have been proposed to address the problem of congestion control at the routers; see [Yang and Reddy 1995], [Haider et al. 2001] and [Zhu et al. 2002]. This includes Random Early Detection (RED) algorithm, which was first proposed in [Floyd and Jacobson 1993]. Later its use was recommended in IP routers for implementation of Active Queue Management (AQM) [Braden et al. 1998]. It has been recently implemented in commercially available routers. It is believed that RED routers will solve the major problems in existing drop-tail routers such as synchronization of flows and will support quality of service by providing intelligent marking of packets. The performance of RED is dependent upon the proper tuning/matching of its four intertwined parameters $\{min_{th}, max_{th}, max_p, w_q\}$ to the three parameters influencing congestion: bottleneck link bandwidth, number of TCP flows and round-trip time.

Central to the RED operation is the Exponential Weighted Moving Average (EWMA) mechanism for controlling the instantaneous queue size that acts as a Low Pass Filter (LPF) to absorb transient traffic bursts. It is used in conjunction with a piecewise linear marking/dropping probability function to get the desired functionality. This paper presents two modifications to the use of EWMA by RED which are shown to improve its performance under sudden change in traffic load from a high to low value. Since the new algorithm combines use of both instantaneous and EWMA queue sizes, it is named Hybrid RED. The performance of the newly developed algorithm is determined by running ns-2 based simulations.

The rest of the report is organized as follows. In Chapter 2 a brief overview of the basic RED algorithms, and a discussion of their queue behavior, is presented. Chapter 3 discusses related work. Standard RED based on EWMA queue size is analyzed in Chapter 4. The Hybrid RED algorithm is derived in Chapter 5. Chapter 6 contains simulation results, performance analysis and conclusions for Hybrid RED.

Chapter 2

Background

We first briefly review the two basic forms of the existing versions of RED algorithms and then discuss how they influence the queue properties.

2.1 RED Algorithm

The basic version of RED uses the following LPF with a constant w_q (called queue weight, or smoothing factor, or forgetting factor), to calculate the EWMA, $\bar{q}$, of the instantaneous queue size q:

$$\bar{q} \leftarrow (1 - w_q) \cdot \bar{q} + w_q \cdot q; \tag{2.1}$$

see [Floyd and Jacobson 1993]. In equation (2.1), w_q determines the time constant of the LPF which in effect determines the length of memory used in the averaging process [Misra et al. 2001]. One can split equation (2.1) into the two components: (a) contribution of the present value of instantaneous queue size q, which is represented by the term $w_q \cdot q$, and (b) contribution of the previous value of EWMA queue size, given by the term $(1 - w_q) \cdot \bar{q}$. It shows that as w_q is increased, the contribution of q increases, or memory decreases, and vice versa. The linear marking/dropping probability function of the basic RED algorithm can be expressed algebraically as:

$$p_b = \begin{cases} 0, & \text{if } \bar{q} < min_{th}; \\ 1, & \text{if } \bar{q} \geq max_{th}; \\ \frac{\bar{q} - min_{th}}{max_{th} - min_{th}} \cdot max_p, & \text{if } \bar{q} \in [min_{th}, max_{th}]. \end{cases} \tag{2.2}$$

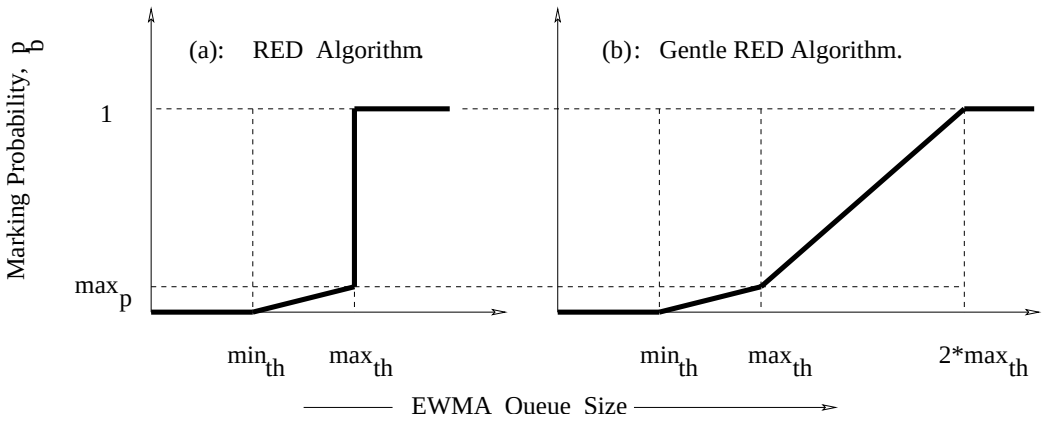

Figure 2.1: Packet marking/dropping probability curves of the basic RED, Fig. 1 (a), and Gentle RED, Fig. 1 (b).

where min_{th}, max_{th} are minimum and maximum thresholds for EWMA queue size, respectively, and max_p is the maximum value of packet mark/drop probability; see Figure 2.1 (a). For a buffer size B, it is generally recommended to use $max_{th} < B$, $max_{th} = 3 \cdot min_{th}$, $max_p = 0.1$ and $w_q = 0.002$, see [Floyd and Jacobson 1993]. The discontinuity in maximum mark/drop probability, max_p, at $\bar{q} \geq max_{th}$ in equation (2.2) may cause violent oscillations in the instantaneous queue size q; see [May et al. 2000] and [Rizzo 1997]. In general, the operation is critically dependent upon the proper selection of four intertwined parameters, $\{min_{th}, max_{th}, max_p, w_q\}$; see [Floyd and Jacobson 1993] and [Floyd 1997] for parameter settings of the basic RED.

2.2 Gentle RED Algorithm

The gentle version of RED was proposed to avoid the sudden jump in probability function in (2.2) by modifying it as follows:

$$p_b = \begin{cases} 0, & \text{if } \bar{q} < min_{th}; \\ 1, & \text{if } \bar{q} \geq 2 \cdot max_{th}; \\ \frac{\bar{q} - min_{th}}{max_{th} - min_{th}} \cdot max_p, & \text{if } \bar{q} \in [min_{th}, max_{th}]; \\ \frac{\bar{q} - max_{th}}{max_{th}} \cdot (1 - max_p) + max_p, & \text{otherwise}; \end{cases} \tag{2.3}$$

see [Floyd 2000]. If intermarking time, X, is equal to number of packets, n, that arrive after a marked/dropped packet (until next packet is marked/dropped) then

$Prob[X = n] = (1 - p_b)^{n-1} \cdot p_b$. It shows that X is a random geometric variable with $E(X) = 1/p_b$, which will mark/drop packets at irregular time intervals and thus leading to global synchronization of TCP flows, Floyd and Jacobson [1993].

Thus, both (2.2) and (2.3) will result in an undesirable geometrical pattern for marking/dropping of incoming packets. Therefore, it is desired that X should be a uniform random variable, which can be achieved by using the following expression:

$$p_a = \frac{p_b}{1 - count \cdot p_b}, \tag{2.4}$$

where $count$ is the number of unmarked packets that have arrived since the last marked/dropped packet; for more details see Floyd and Jacobson [1993]. It is clear that the operation of both versions of RED algorithm, as given by (2.2) and (2.3), is crucially dependent on the use of the EWMA queue size $\bar{q}$. In a later section it will be shown that $\bar{q}$ is not by itself an ideal indicator of congestion events in router and thus leads to non-optimized performance, due to unnecessary packet marking/dropping.

The instantaneous queue size, q, in RED routers varies between zero and the buffer size B. When q hits zero, a period of under-utilization of the bottleneck link bandwidth begins. On the other hand, when q reaches the upper limit B, all of the incoming packets will be dropped, and such forced drops will require retransmissions. This process will lead to congestion. Transient periods of bursty traffic can cause wide fluctuations of q, which can be diminished by introducing the EWMA mechanism.

2.3 EWMA Tuning

The use of EWMA for packet marking/dropping in conjunction with a simple linear function requires optimal value of queue weight, w_q [Young 1984] and [Floyd and Jacobson 1993]. If w_q is too large then transient congestion will not be filtered out and if it is too small then $\bar{q}$ will respond too slowly to enforce the desired changes in p_b. Larger values of w_q also encourage oscillations in the queue size. Assuming that the queue was initially empty and it has grown to an instantaneous size of L packets, i.e. $q = L$, over the arrival of L consecutive packets the following expression of EWMA queue size, $\bar{q}$, has been obtained

$$\bar{q} = L + 1 + \frac{(1 - w_q)^{L+1} - 1}{w_q}, \tag{2.5}$$

see [Floyd and Jacobson 1993]. The upper limit for w_q is derived from $\bar{q} < min_{th}$, i.e.:

$$L + 1 + \frac{(1 - w_q)^{L+1} - 1}{w_q} < min_{th}. \tag{2.6}$$

The lower limit for w_q is computed by the simple observation that it takes $\frac{-1}{\ln(1-w_q)}$ packet arrivals to change $\bar{q}$ from 0 to 0.63 while q stays at 1 packet. As stated in the previous subsection, for RED type algorithms the value of w_q is usually chosen small, of the order of 0.002, so as to avoid the bias against sudden bursts of incoming packets.

In [Sirisena et al. 2002] a good value of w_q for a bottleneck link of capacity C and round trip time RTT, was determined as:

$$w_q = 1 - e^{\frac{-1}{10 \cdot C \cdot RTT}} \approx \frac{1}{10 \cdot C \cdot RTT}, \tag{2.7}$$

which is based on a control-theoretic analysis of RED type algorithms; and is a general form of the expression given in [Floyd et al. 2001]. Using (2.7), the value of w_q can be automatically adjusted according to the network scenario.

The effects of using longer memory (smaller values of w_q in equation (2.1) were further investigated in [Misra et al. 2001] by using simulation studies. It was found that if w_q is decreased beyond a certain critical value then the performance of the RED algorithm degrades as the coefficient of variation of queue occupancy increases. Also, the use of a longer memory in the EWMA process and an increase in the round trip time delay in feedback loop of TCP/IP connections often drive a stable process into oscillatory mode, which is inconsistent with the theoretical results obtained in [Ott 1999]. It is suggested that generally w_q should be chosen in the range of 0.5 to 1 (note that w_q=1 stands for case when $\bar{q}(k) = q(k)$, i.e. no averaging by EWMA process occurs) rather than the current practice of using values in the range of 0.001 to 0.002. However, it is also predicted that for queue size in the buffers of backbone routers, using the RED algorithm with the small values of w_q (0.001 to 0.002) might not be practically detrimental, [Misra et al. 2001]. In this work we use the value $w_q = 0.002$, as has been recommended in [Floyd 1997] and employed in [Floyd et al. 2001]. In next Chapter we present related work to Hybrid RED.

Chapter 3

Related Work

In [Ott 1999], the EWMA process is modeled as *Ornstein-Uhlenbeck* process with delayed feedback which proves that for sufficiently long delay in feedback the process becomes unstable. It is conjectured that widespread use of the EWMA process, at routers equipped with RED-type algorithms, might compromise stability of the Internet. In order to offset these problems, use of both instantaneous and EWMA queue size for packet mark/drop has been suggested literature, see [May et al. 2000].

V. Jacobson and co-workers have suggested that the gain of the filter or queue weight w_q should be different for building queues and draining queues [Jacobson et al. 1999]. For building queues, the response of RED to a unit step function is investigated with w_q chosen to be equal to the inverse of the number queue samples in a round trip time, so that the filter approximates the mean over a Round Trip Time (RTT). The analysis shows that it takes 2 RTTs for RED to reach 90% of the persistent queue level. For draining queues, they suggest that w_q be set to one, i.e. EWMA queue size should be equal to the instantaneous queue size, thus, reinforcing the results reported in [Ott 1999]. Despite of improvements in [Floyd and Jacobson 1993], the trade-off between ability to absorb transient burst of packets and to follow the instantaneous queue size, still exists. More work is required to investigate the effects of persistent queue and the selection of queue sampling interval.

In [May et al. 2000], the use of both instantaneous and EWMA queue size for packet marking decisions was investigated on a testbed. In one of the tests, instantaneous queue size was used in a gentle RED version [Floyd 2000]. These experiments showed that EWMA hinders the anticipation of congestion and better congestion indicators (like instantaneous queue size) are needed to detect incipient

congestion in a network.

Also, it has been pointed out in [Rizzo 1997] that RED algorithm needs modification by employing both q and $\bar{q}$ for packets mark/drop decisions. The following major drawbacks are inherent in RED type algorithms due to the use of only EWMA queue size:

- Since EWMA queue size is computed at each packet arrival, it can grow at any rate depending on the arrival rate. On the other hand the maximum decrease rate in the EWMA queue size is limited by the bottleneck bandwidth (or even less, since recomputation is only triggered by packet arrivals, not departures). Thus EWMA shows averaging of packet arrival rate rather then the true buffer occupancy.

- During packet bursts, the packet arrival rate is greater than the packet departure rate at the bottleneck routers. Due to this difference in the two rates, equation (2.1) resembles more a peak detector than EWMA filter. Hence it has a bias towards large values of queue size.

Such problems in the EWMA queue size are also considered in [Lin and Morris 1997] and a new algorithm, named Flow Random Early Detection (FRED), has been proposed. However, FRED has several performance limitations [Stoica et al. 1998], and also it maintains state variables for all active flows which makes it less scalable.

The LPF/ODA algorithm, [Zheng and Atiquzzaman 2001] and [Zheng and Atiquzzaman 2002], aims to improve link utilization when EWMA is at a high value ($min_{th} < \bar{q} < max_{th}$ *or* $\bar{q} > max_{th}$) but the instantaneous queue size is lower than min_{th}. The algorithm can be summarized as follows:

- During normal operation the queue size is controlled by EWMA value, which is the default control mechanism of RED algorithm. During this phase, called *Active Drop Phase or LPF*, the router will behave as normal RED.

- If the instantaneous queue size stays below min_{th} for three consecutive packet arrivals, then it indicates the onset of the *Over Drop Avoidance (ODA) Phase* during which the EWMA is reduced to half of its present value.

- After this reduction, updating of the EWMA queue size is continued as in normal RED.

In [Zheng and Atiquzzaman 2002], number of ODA-phase triggering packets, $\mathcal{N}_o$, and amount of reduction in EWMA queue size (during that ODA-phase) has been arbitrarily chosen as 3 and 0.5, repectively. In contrast, this paper determines the effects of changes in number of ODA triggering packets and amount of reduction in EWMA queue size, on throughput and loss rate at the bottleneck link while employing the Hybrid RED. Furthermore, the results presented in both [Zheng and Atiquzzaman 2001] and [Zheng and Atiquzzaman 2002] provide no information about error statistics. Whereas, this paper reports the results for Hybrid RED obtained after multiple runs of ns-2 based simulations.

In general, our work supports the concept of employing both instantaneous and EWMA queue size during packet marking/dropping decisions in RED type algorithms which is in line with the findings of [Rizzo 1997], [Ott 1999], [Jacobson et al. 1999], [May et al. 2000], [Zheng and Atiquzzaman 2001] and [Zheng and Atiquzzaman 2002].

Chapter 4

Analysis of RED with EWMA queue size

Consider a bottleneck link of capacity C being served by a RED router, having a small value of w_q, at a rate of μ where $\mu < C$. Thus, the instantaneous queue size, q, has not built up and the EWMA queue size, $\bar{q}$, is small in value. Now, let another traffic flow join the session, making the aggregate packet arrival rate at router ports is greater than C. Now q is large and $\bar{q}$ is computed on packet arrivals, which makes $\bar{q}$ to grow much larger than max_{th}, causing all of the new incoming packets to be dropped.

This dropping of packets will continue for a long time even after the queue has become empty, because $\bar{q}$ will take a long time to return below max_{th}. During this epoch of time, the transmission rate of packets from the queue will reach a very low value as almost all of the incoming traffic will be dropped. Such cases can be identified in Figs.3 and 9 of [Floyd and Jacobson 1993]. The situation will momentarily return to normal after the value of $\bar{q}$ drops below max_{th}. However the consequent increase in sending rate will cause $\bar{q}$ to increase again. Thus, we will reach a state in which $\bar{q}$ oscillates around max_{th} and packets will be again dropped forcibly [Floyd and Fall 1997]. Hence, depending upon the rate of incoming packets, bottleneck link rate, and responsiveness of flows, there might be epochs when all of the packets will be discarded while the link is idle.

Two such cases are illustrated in Figures 4.1 and 4.2, where q is well below min_{th} but $\bar{q} > max_{th}$ and $min_{th} < \bar{q} < max_{th}$ respectively, causing unwanted dropping of packets. It indicates that EWMA by itself does not always provide a true picture of the backlog in buffers of RED queues. There can be instants when EWMA gives false indications of buffer occupancy which will lead to wastage of

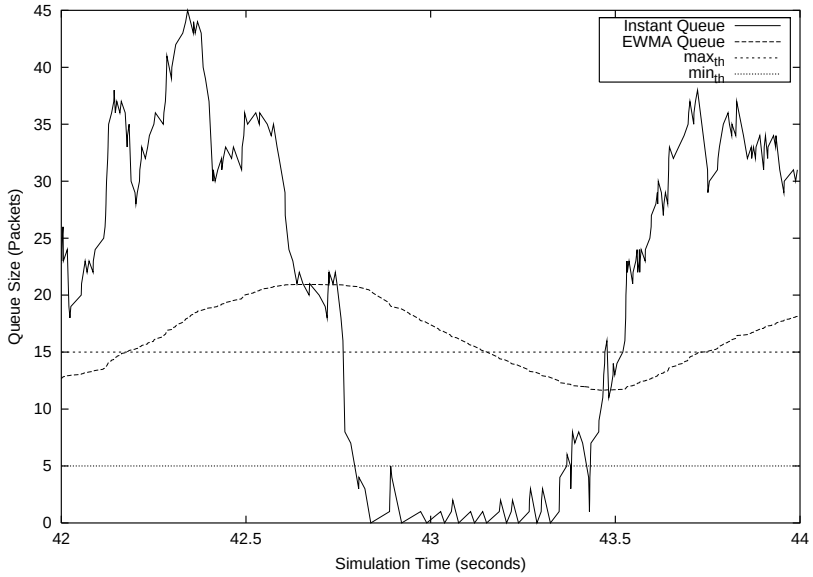

Figure 4.1: Queue snapshot when $q < min_{th}$ and $\bar{q} > max_{th}$

precious network bandwidth. Thus, in the following section we present a new al-

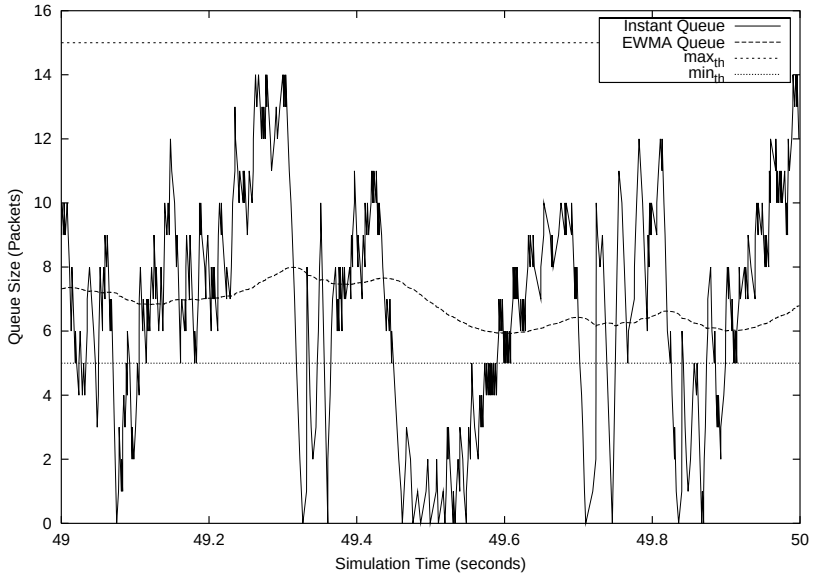

Figure 4.2: Queue snapshot when $q < min_{th}$ and $min_{th} < \bar{q} < max_{th}$

gorithm, called Hybrid RED, aimed at overcoming the above mentioned problems by utilizing both instantaneous queue size, q, and its EWMA value, $\bar{q}$.

Chapter 5

Hybrid RED

The new Hybrid RED algorithm adjusts $\bar{q}$ during $q < min_{th}$ periods and also takes the value of q into account during enqueue operation of incoming packets. It is derived as follows.

5.0.1 Decreasing $\bar{q}$

Discretizing equation (2.1) at time instant k and converting the assignment operation to equality yields the following equation:

$$\bar{q}(k) = (1 - w_q) \cdot \bar{q}(k-1) + w_q \cdot q(k); \tag{5.1}$$

see [Hostetter 1988]. By rearranging terms in equation (5.1) we get:

$$q(k) = \left(\frac{1 - w_q}{w_q}\right) \cdot \bar{q}(k-1) + \frac{\bar{q}(k)}{w_q}. \tag{5.2}$$

Equations (5.1) and (5.2) indicate the interdependency of q and $\bar{q}$ created through the use w_q in EWMA process given in (2.1).
As $q(k)$ varies between 0 and B, it also traverses the regions between 0 to min_{th} and between min_{th} and max_{th}, see Fig 2.1. If $q(k) \leq min_{th}$ for time intervals greater than a single packet arrival time (or, for $\theta \geq 1$ consecutive packet arrivals), then we need to decrease the first term on the RHS of equation (5.1) by a factor ξ so that $\bar{q}$ is either less than min_{th} or close to it. In discrete terms, we can write this sub-algorithm as:

$$\bar{q}(k) = \begin{cases} (1 - w_q) \cdot \bar{q}(k-1) + w_q \cdot q(k), & \text{for } q > min_{th}; \\ \left(\frac{1-w_q}{\xi}\right) \cdot \bar{q}(k-1) + w_q \cdot q(k), & \text{for } q < min_{th},\ \theta > 1. \end{cases} \tag{5.3}$$

The usefulness of equation (5.3) depends upon the proper choice of the pair $\{\theta, \xi\}$, otherwise it can degrade the performance of Hybrid RED and even cause stability problems by causing wide variations in $\bar{q}$. Thus there must be a balance between the values of ξ and θ. For large values of θ, we should choose larger values of ξ and for smaller values of θ, ξ must be smaller.
If ξ is chosen such that $min_{th} < \bar{q} < max_{th}$ while $q(k) \leq min_{th}$, then we will still have some probabilistic packet drops as governed by equations (2.2) or (2.3). These drops will be fewer in number and thus will cause less bandwidth wastage than in the previous state of $\bar{q} > max_{th}$.

5.1 Packet marking/dropping based on q and $\bar{q}$.

The enqueuing and marking/dropping instances of Hybrid RED as function of q and $\bar{q}$ are shown in Figure 5.1 divided into regions labeled from 1 to 11. Regions 1 and 11 correspond to forced packet dropping and 2, 3, 4, 6, 7 correspond to probabilistic or random packet marking/dropping, [Floyd and Fall 1997]. The RED type algorithms derived from either its basic or gentle version do forced marking/dropping of packets in all regions for which $\bar{q} > max_{th}$ (i.e. regions 4,7 and 10). By contrast, in Hybrid RED we enqueue packets in regions 7 and 10 which are ($min_{th} < q < max_{th}$) and ($q < min_{th}$), respectively. A more conservative approach would be to use region 10 only and not use region 7 for enqueuing operations while $\bar{q} > max_{th}$. Also skipping of region 10 will turn this sub-algorithm OFF and we will default to RED regions only.

5.2 Pseudocode of Hybrid RED

The sub-algorithms described in the previous two subsections can be combined together into the following composite and coherent Hybrid RED algorithm:

Algorithm 1 Hybrid RED algorithm

procedure COMPUTEEWMA(w_q, $q(k)$, $\bar{q}(k-1)$)
  for all k **do**
    $\bar{q}(k) \leftarrow (1 - w_q) \cdot \bar{q}(k-1) + w_q \cdot q(k)$
  end for
end procedure
procedure COMPUTEPROBABILITY(min_{th}, max_{th}, max_p, $\bar{q}(k)$)
  if $\bar{q}(k) < min_{th}$ **then**
    $p_b \leftarrow 0$
  else if $\bar{q}(k) > 2 * max_{th}$ **then**
    $p_b \leftarrow 1$
  else if $\bar{q}(k) \in [min_{th}, max_{th}]$ **then**
    $p_b \leftarrow \frac{\bar{q}(k) - min_{th}}{max_{th} - min_{th}} \cdot max_p$
  else
    $p_b \leftarrow \frac{\bar{q}(k) - max_{th}}{max_{th}} \cdot (1 - max_p) + max_p$
  end if
end procedure
procedure UNIFORMLYDISTRIBUTEPROBABILITY(p_b, count)
  $p_a \leftarrow \frac{p_b}{1 - p_b \cdot count}$
end procedure ▷ RED's procedures completed
procedure REDUCEEWMA($\bar{q}(k), \bar{q}(k-1), \theta, min_{th}, \xi$)
  for θ consecutive packets **do**
    if $\bar{q}(k) < min_{th}$ **then**
      $\bar{q}(k) \leftarrow \frac{\bar{q}(k-1)}{\xi}$
    else
      Proceed as normal RED
    end if
  end for
end procedure
procedure ENQUEUEPACKETS($q(k), \bar{q}(k), min_{th}, max_{th}$)
  if $(\bar{q}(k) > max_{th}) \&\& (\bar{q}(k) < 2 * max_{th}) \&\& (q(k) < min_{th})$ **then**
    Enqueue Packet
  else if $(\bar{q}(k) > max_{th}) \&\& (min_{th} < q(k) < max_{th})$ **then**
    Enqueue Packet
  else
    Proceed as normal RED
  end if
end procedure

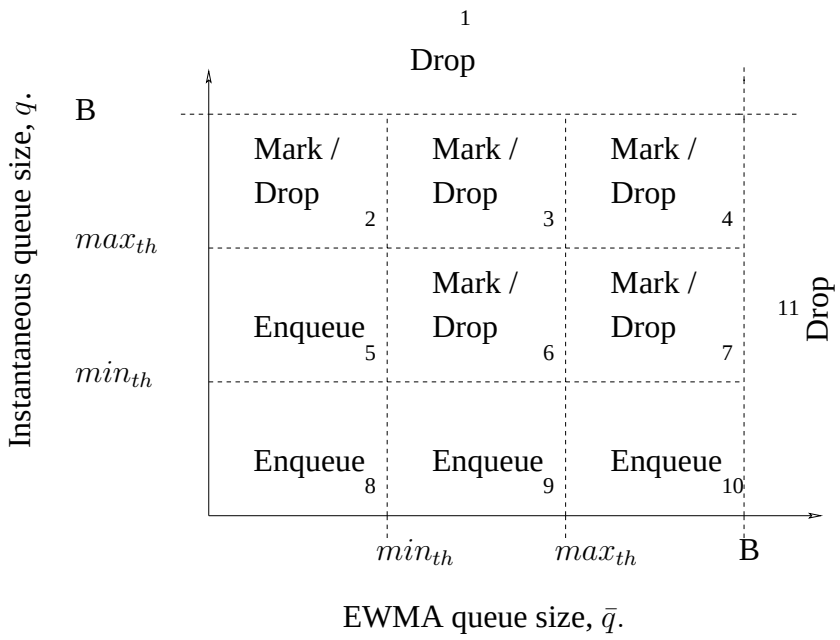

Figure 5.1: Packet enqueuing and marking/dropping logic for Hybrid RED algorithm.

Chapter 6

Performance Evaluation

Network simulator ns-2.27 [DARPA/VINT et al. 2004] has been used to evaluate performance of Hybrid RED algorithm by comparison with RED, Adaptive RED and LPF/ODA algorithms. In the following section we describe two network topologies which have been used in simulation experiments.

6.1 Simulation Scenarios

The first simulation scenario is illustrated in Figure 6.1, which consists of a simple dumbbell topology consisting of three main links, a-d, b-d and c-d, each having capacity of 100 Mbps and propagation delays of 1, 3 and 5 ms, respectively. The bottleneck link d-e has a capacity of 10 Mbps with a delay of 5 ms. Each of the nodes a, b and c is sending FTP traffic during the whole simulation period (50 s). Thus, it represents a network operating under fixed traffic conditions.

The second simulation topology is shown in Figure 6.2, Floyd et al. [2001]. It consists of four main links (labelled as a-c, b-c, e-d and f-d) each having a capacity of 10 Mbps and delays of 2, 3, 4 and 5 ms, respectively. The bottleneck link, c-d, has a capacity 1.5 Mbps with a delay of 20 ms. The forward traffic consists of two long lived TCP flows (from a to e and b to f) and the reverse traffic consists of one long lived TCP flow (from e to b). The presence of reverse data traffic will cause ACK compression (loss of ACK packets) which will increase the burstiness in data traffic on the forward paths. At time of 25 s twenty new TCP flows start one after the other at intervals of 0.1 s. This models a sharp change in the level of contention among competing users. Thus, this network represents a dynamic traffic scenario.

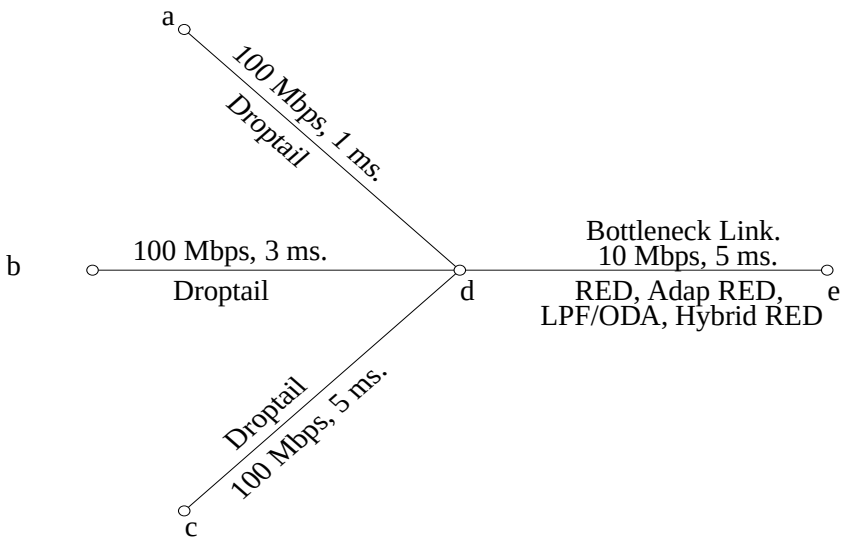

Figure 6.1: Network topology 1.

The bottleneck link is using one of the algorithms: RED [Floyd and Jacobson 1993], Adaptive RED [Floyd et al. 2001], LPF/ODA [Zheng and Atiquzzaman 2002] and Hybrid RED router and all other links are using Droptail routers. Packet size is 1000 bytes and buffer size, B, of all routers is 50 packets. We use $min_{th} = 5$ and $max_{th} = 15$ packets and $w_q = 0.002$ for Hybrid RED, which are also the default values of RED and LPF/ODA algorithms. Thus, router queues are ensured to be under active queue management as in [Zheng and Atiquzzaman 2002]. We use all algorithms in dropping rather in packet marking mode as we are not employing Explicit Congestion Notification, [Ramakrishnan et al. 2001].

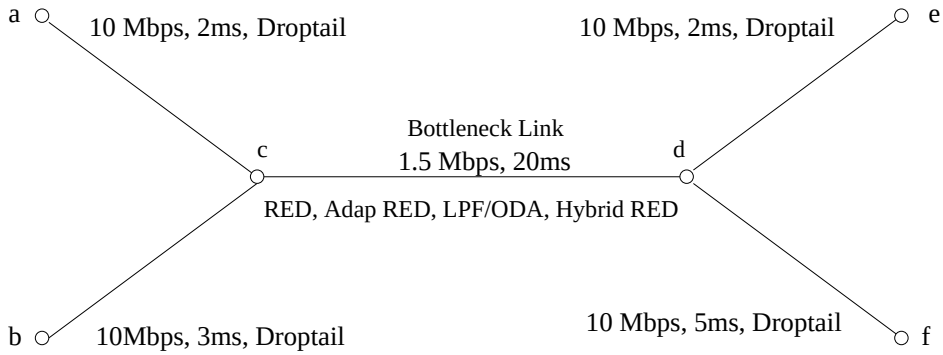

Figure 6.2: Network topology 2.

6.2 Performance Metrices

Performance metrics used for evaluation of Hybrid RED are loss rate (δ), link utilization (η) and EWMA of instantaneous queue size variations. Values of δ and η are computed from following relations:

$$\delta = \frac{\sum Packets\ Dropped\ at\ Queue}{\sum Packets\ Enqueued}; \tag{6.1}$$

$$\eta = \frac{\sum Packets\ sent\ over\ bottleneck\ link}{Maximum\ Packets\ sent\ on\ bottleneck\ link}; \tag{6.2}$$

see [Feng et al. 1999]. For plotting purposes, we compute mean values and associated errors for loss rate and link utilization, δ_{av} and η_{av}, for multiple run of simulations.

6.3 Simulation Results

We performed simulations on the networks shown in Figures 6.1 and 6.2. We employed RED, LPF/ODA and Hybrid RED algorithms at the bottleneck link of both networks. We employed RED, LPF/ODA and Hybrid RED algorithms at the bottleneck link of both networks. In order to operate Hybrid RED algorithm in an optimal fashion, we need to select suitable values of θ and ξ. In [Zheng and Atiquzzaman 2001] [Zheng and Atiquzzaman 2002] the values of θ and ξ are arbitrarily chosen as 3 and 2, respectively. However, this work presents a spectrum of simulation results for different combinations of θ and ξ for the Hybrid RED algorithm.

A quantitative comparison (for both networks) between mean link utilization (η_{av}) with associated error ($\Delta_{\eta_{av}}$) and mean loss rate (δ_{av}) with associated error ($\Delta_{\delta_{av}}$) for RED, Adaptive RED and LPF/ODA algorithms, after multiple runs (100) of simulation, are presented in Table 6.1. It shows that for the static scenario, Figure 6.1, LPF/ODA has a lower link utilization but also a lower loss rate as compared to both RED and adaptive RED. Whereas for the dynamic scenario, Figure 6.2, LPF/ODA has the highest link utilization and also the lowest loss rate, as compared to RED and its adaptive version. In the case of Hybrid RED these values depend upon the selection of ξ and θ. Further we perform simulations with proposed Hybrid RED algorithm.

Employing Hybrid RED at the bottleneck link of the network shown in Figures 6.1 and 6.2, we performed simulations for all combinations of $\theta = \{1, 2, 3, 4, 5\}$

Table 6.1: A comparison between RED, Adaptive RED and LPF/ODA algorithms. The values of $\bar{\eta}$, Δ_η, $\bar{\delta}$ and Δ_δ are in %.

Algorithm	Topology 1 (Fig. 6.1)				Topology 2 (Fig. 6.2)			
	η_{av}	$\Delta_{\eta_{av}}$	δ_{av}	$\Delta_{\delta_{av}}$	η_{av}	$\Delta_{\eta_{av}}$	δ_{av}	$\Delta_{\delta_{av}}$
RED	98.73	0.23	1.02	0.09	90.38	3.42	8.15	6.76
Adapt RED	95.38	0.36	0.97	0.06	88.93	2.22	8.25	6.75
LPF/ODA	94.93	1.04	0.84	0.18	36.67	2.04	7.93	6.80

Table 6.2: Performance of Hybrid RED for $\theta = 1$ packet.

Hybrid RED (θ, η)	Topology 1 (Fig. 6.1)				Topology 2 (Fig. 6.2)			
	η_{av}	$\Delta_{\eta_{av}}$	δ_{av}	$\Delta_{\delta_{av}}$	η_{av}	$\Delta_{\eta_{av}}$	δ_{av}	$\Delta_{\delta_{av}}$
(1, 1.10)	98.96	0.79	0.87	0.11	89.25	2.56	8.13	6.79
(1, 1.25)	98.92	0.82	0.79	0.15	89.80	2.40	8.12	6.69
(1, 1.50)	98.86	0.90	0.77	0.14	90.24	1.73	8.10	6.64
(1, 2.00)	98.87	1.01	0.77	0.18	90.29	3.23	8.07	6.69
(1, 3.00)	98.86	0.97	0.77	0.19	90.53	2.03	8.12	6.80

Table 6.3: Performance of Hybrid RED for $\theta = 2$ packets.

Hybrid RED (θ, η)	Topology 1 (Fig. 6.1)				Topology 2 (Fig. 6.2)			
	η_{av}	$\Delta_{\eta_{av}}$	δ_{av}	$\Delta_{\delta_{av}}$	η_{av}	$\Delta_{\eta_{av}}$	δ_{av}	$\Delta_{\delta_{av}}$
(2, 1.10)	98.93	0.95	0.88	0.13	89.30	2.03	8.06	6.67
(2, 1.25)	98.87	1.20	0.80	0.13	89.67	2.34	8.10	6.87
(2, 1.50)	98.82	1.23	0.77	0.17	90.23	2.32	8.07	6.65
(2, 2.00)	98.83	0.87	0.77	0.14	90.24	1.95	8.10	6.75
(2, 3.00)	98.78	1.38	0.77	0.17	90.39	2.00	8.09	6.76

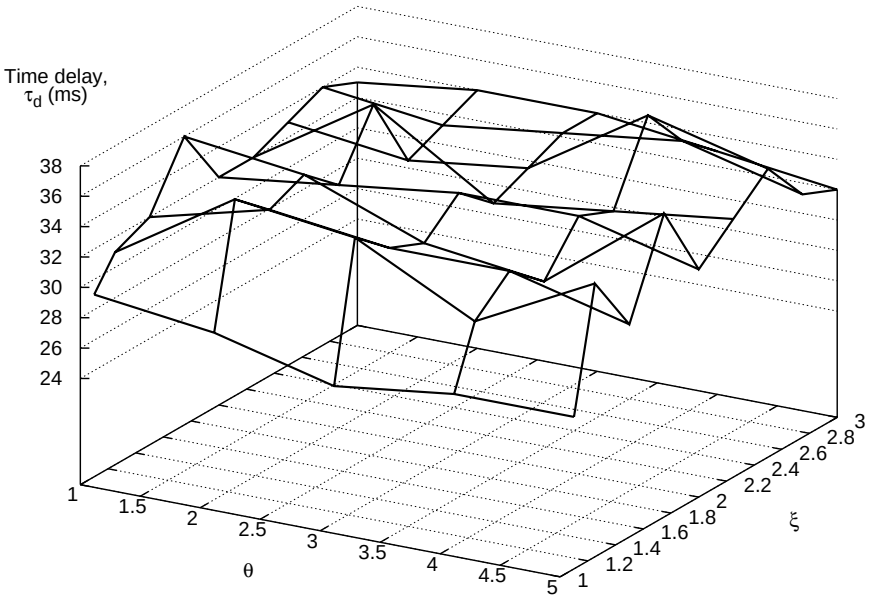

Figure 6.3: Variations in time delay for Hybrid RED for simulation topology shown in Fig. 6.1.

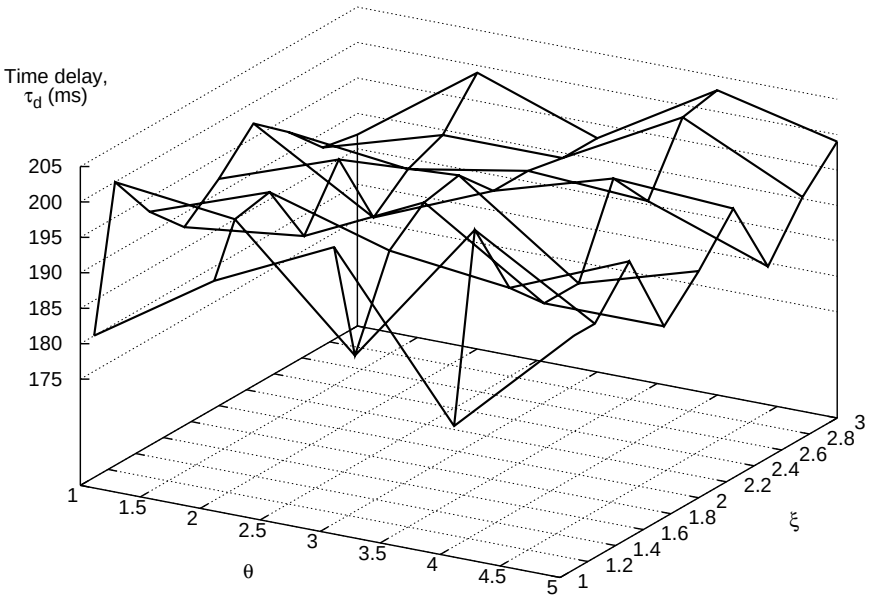

Figure 6.4: Variations in time delay for Hybrid RED for simulation topology shown in Fig. 6.2.

Table 6.4: Performance of Hybrid RED for $\theta = 3$ packets.

Hybrid RED (θ, η)	Topology 1 (Fig. 6.1)				Topology 2 (Fig. 6.2)			
	η_{av}	$\Delta_{\eta_{av}}$	δ_{av}	$\Delta_{\delta_{av}}$	η_{av}	$\Delta_{\eta_{av}}$	δ_{av}	$\Delta_{\delta_{av}}$
(3, 1.10)	98.90	0.88	0.89	0.14	89.26	2.66	8.06	6.94
(3, 1.25)	98.84	0.89	0.81	0.15	89.71	2.59	8.07	6.68
(3, 1.50)	98.79	0.89	0.78	0.14	90.19	1.87	8.05	6.72
(3, 2.00)	98.74	0.94	0.77	0.17	90.22	2.16	8.06	6.61
(3, 3.00)	98.77	1.10	0.77	0.17	90.38	2.21	8.08	6.97

and $\xi = \{1.1, 1.25, 1.5, 1.75, 2.0, 2.5, 3.0\}$. The resulting link loss rate, δ, and link utilization, η, for a single run of simulation are plotted in Figures 6.5 and 6.6, respectively, which are not very smooth curves. Therefore, we further perform multiple runs (100 times with different independent streams of random numbers) of each simulation and computed minimum, maximum and mean values of link loss and link utilization, i.e. δ_{av} and η_{av}. These results are presented below as Experiment 1 and 2.

Experiment 1

For network shown in Figure 6.1 (static scenario) the average loss rate of REDand LPF/ODA are 1.02 ± 0.045 % and 0.84 ± 0.09 %, respectively, which are plotted in Figure 6.7. For Hybrid RED the loss rate for different values of θ has been plotted in Figures 6.8 to 6.12. A comparison of average loss rate at bottleneck link, δ_{av}, for Hybrid RED, using different values of θ is presented in Figure 6.13. It shows that link loss rate starts decreasing with increase in ξ, i.e. decrease from 1.016 % to values close to 0.8 % which is less than other algorithms, i.e. RED and LPF/ODA.

The average link utilization, η_{av}, for RED, and LPF/ODA are 98.73 ± 0.11 % and 94.93 ± 0.02 %, respectively, which are plotted in Figure 6.14. Results of multiple run simulations for η_{av} values for Hybrid RED are shown in Figures 6.15 to 6.19 with associated limit of errors. A comparison between link utilization, for different values of θ, is shown in Figure 6.20. It shows that, with an increase in ξ, average link utilization first starts increasing untill a certain maximum value and then starts to decrease, i.e. first an increase from 98.7 % to 98.97 % and next it decreases to lower values. Therefore, until $\xi < 1.1$ a tradeoff exists between lower average loss rate and higher average link utilization and for $\xi > 1.1$ both decreases

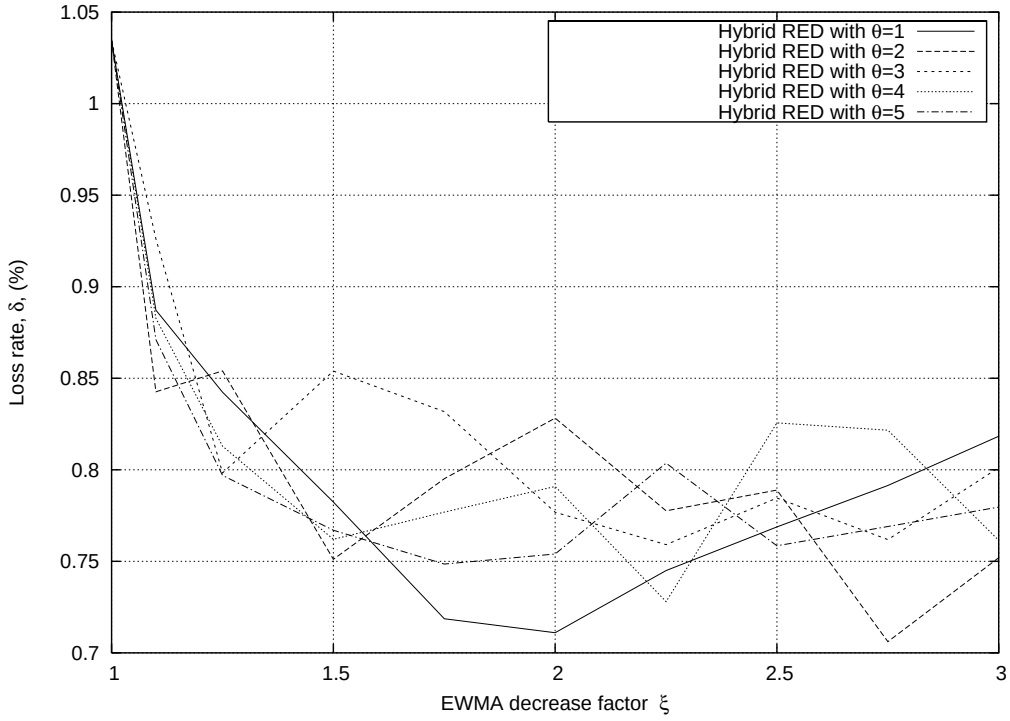

Figure 6.5: Link loss rate (single run) with Hybrid RED in simulation topology shown in Figure 6.1.

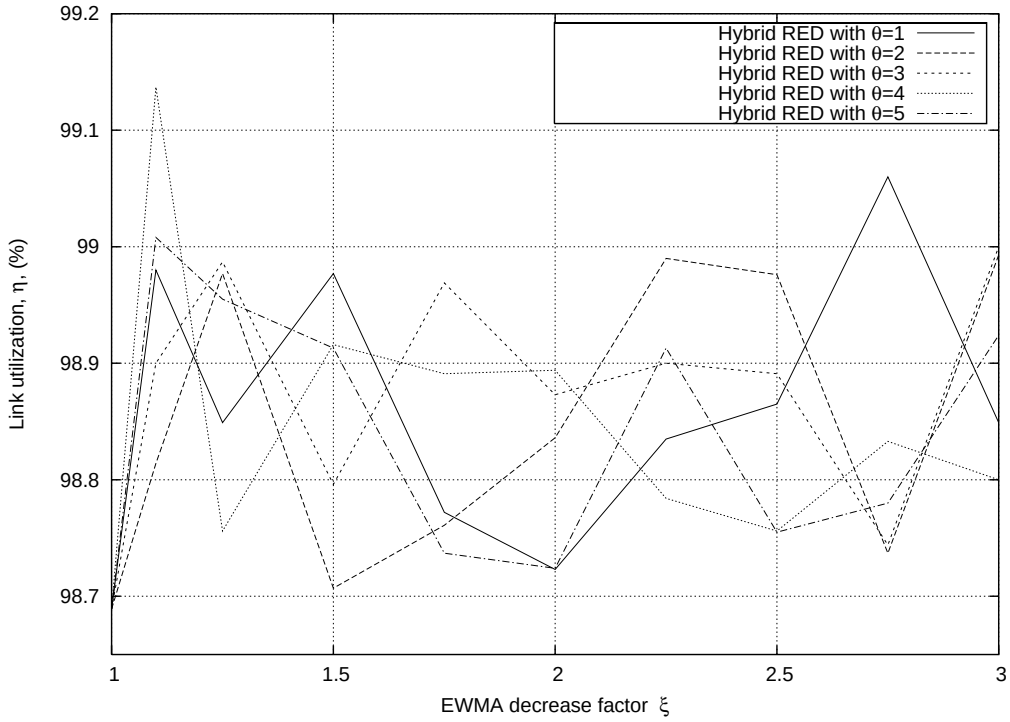

Figure 6.6: Link utilization (single run) with Hybrid RED in simulation topology shown in Figure 6.1.

simultaneously. Hence, by proper selection of ξ and θ the Hybrid RED algorithm can provide lower link loss rate as well as higher link utilization as compared to RED and LPF/ODA algorithms.

However, it is important to consider the effects of variations in ξ and θ on stability of EWMA queue size. Towards this end, EWMA queue size for RED, and LPF/ODA are shown in Figures 6.21 and 6.22 respectively, whereas for Hybrid RED it is shown in (with $\theta = \{1, 2, 3, 4, 5\}$ and $\xi = \{1.25, 2.0\}$) in Figures 6.23 to 6.32. We have also shown the upper and lower thresholds for EWMA queue size (i.e. $max_{th} = 15$ and $min_{th} = 5$ packets) on all EWMA queue size plots. EWMA queue size for RED, Figure 6.21, lies between $max_{th} = 15$ and min_{th} and for LPF/ODA, Figure 6.22, it is fluctuating approximately between 1 and 12 packets, for whole of the simulation time. Whereas, for Hybrid RED there is lesser fluctuation for smaller values than for larger values of ξ.

Experiment 2

Next, we perform simulations on the topology shown in Figure 6.2 (dynamic scenario). In this case we employ RED, Adaptive RED, LPF/ODA and Hybrid RED algorithm at the bottleneck link. In order to compare results, we have used same set of combinations of values of θ and ξ as in the case of Figure 6.1, i.e. of $\theta = \{1, 2, 3, 4, 5\}$ and $\xi = \{1.1, 1.25, 1.5, 1.75, 2.0, 2.5, 3.0\}$. The average loss rate for RED, Adaptive RED and LPF/ODA is shown in Figure 6.33, whereas for Hybrid RED it has been shown in Figures 6.34 to 6.38. A comparison between loss rates, for different values of θ, has also been shown in Figure 6.39. In this case, with an increase in ξ, the average loss rate starts to decrease from 8.21 to a value close to 8.1 %.

The average link utilization for RED, Adaptive RED and LPF/ODA is shown in Figure 6.40, whereas for Hybrid RED it has been shown in Figures 6.41 to 6.45. A comparison between link utilization for different values of θ has also been shown in Figure 6.46. It shows that average link utilization starts to increase from 88.18 % to a value close to close to 90.5 %.

In contrast to previous static case, Figure 6.1, the average link utilization in this dynamic case does not decrease with an increase in ξ. Thus, in a dynamic scenario, higher values of ξ are required for attaining higher values of η_{av}. Also, a tradeoff between average loss rate and average link utilization is absent in this case. However, η_{av} increases to maximum value for $\theta = 1$ as in the static case. Hence, it is proposed that $\theta = 1$ and $\xi < 1.5$ should be chosen for nearly optimal operation of Hybrid RED as the average loss rate and average link utilization curves level out for $\xi > 1.5$. EWMA queue size for RED, Adaptive RED and LPF/ODA has been plotted in Figures 6.47, 6.48 and 6.49, respectively. For Hy-

brid RED we plot EWMA queue size in Figures 6.50 to 6.59. The EWMA plots for RED and Adaptive RED are similar to those obtained in [Floyd et al. 2001]. It shows better performance (convergence to target level) of Adaptive RED as compared to both RED and LPF/ODA algorithms. For Hybrid RED, the magnitude of abrupt fluctuations in EWMA queue size depends upon choice of θ and ξ, which is less than that of LPF/ODA.

6.4 Conclusions

The limitations of using only the EWMA queue size for packet marking/dropping in RED type algorithms at time of congestion are discussed. A Hybrid RED algorithm is proposed which combines EWMA queue size with the instantaneous queue size for making packet marking/dropping decisions and which also incorporates the adjustments to the EWMA queue size when $q < min_{th}$ over a certain number of consecutive packet arrivals. We experimented with reduction in EWMA queue size when $q < min_{th}$ for 1 to 5 packet arrivals and the results are presented.

In order to study the queue dynamics, we have plotted EWMA queue size graphs for networks with both static and dynamic traffic scenarios. Generally, for higher values of ξ, the EWMA queue size becomes noisy and thus fluctuates rapidly, which might cause queue stability problems. On the other hand, higher values of ξ improve bottleneck link loss rate as seen in Figures 6.5 and 6.13. There seems to be a trade-off between stable queue behavior and better link utilization, as well as lower link loss rate.

These simulation studies show that for a static traffic scenario the new Hybrid RED algorithm gives a better loss rate and link utilization as compared to existing RED and LPF/ODA algorithms. In the case of dynamic traffic scenarios the performance of Hybrid RED depends upon the choice of θ and ξ. It is suggested that more simulation and analytical studies be done in this regard to explore the optimum values of ξ and θ for different topologies and they should be tied to other RED parameters i.e. $\{min_{th}, max_{th}, max_p, w_q\}$. In the future we intend to study Hybrid RED under different types of traffic mixes such as exponential ON/OFF traffic over UDP with TCP, and also under Explicit Congestion Notification (ECN) based packet marking.

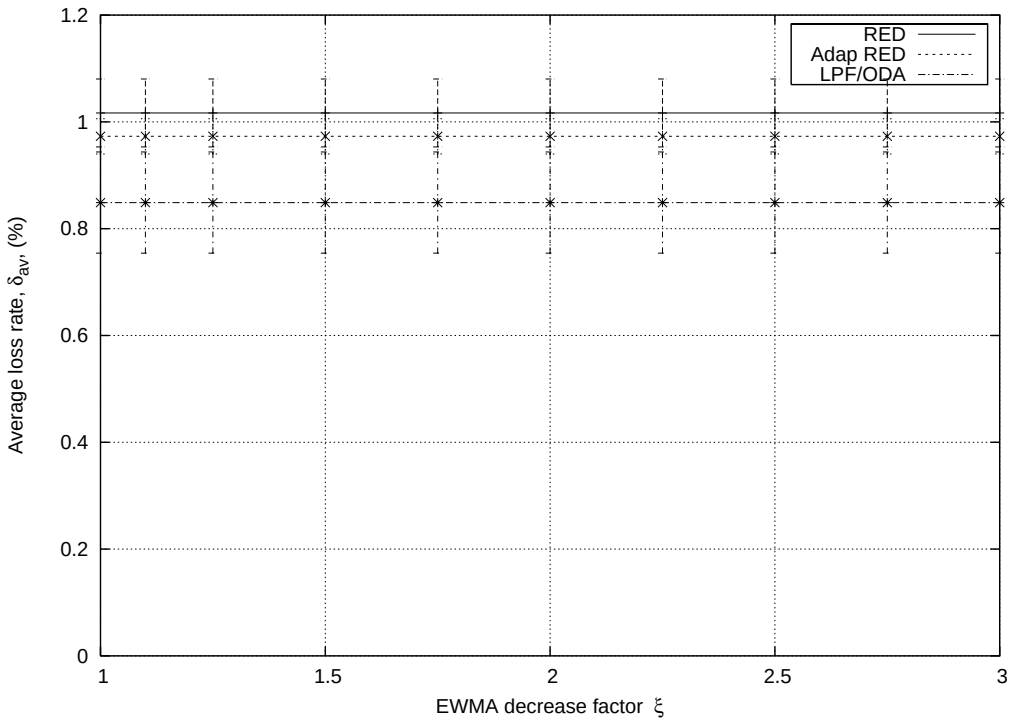

Figure 6.7: Average link loss rate with RED, Adaptive RED and LPF/ODA in the simulation topology shown in Figure 6.1.

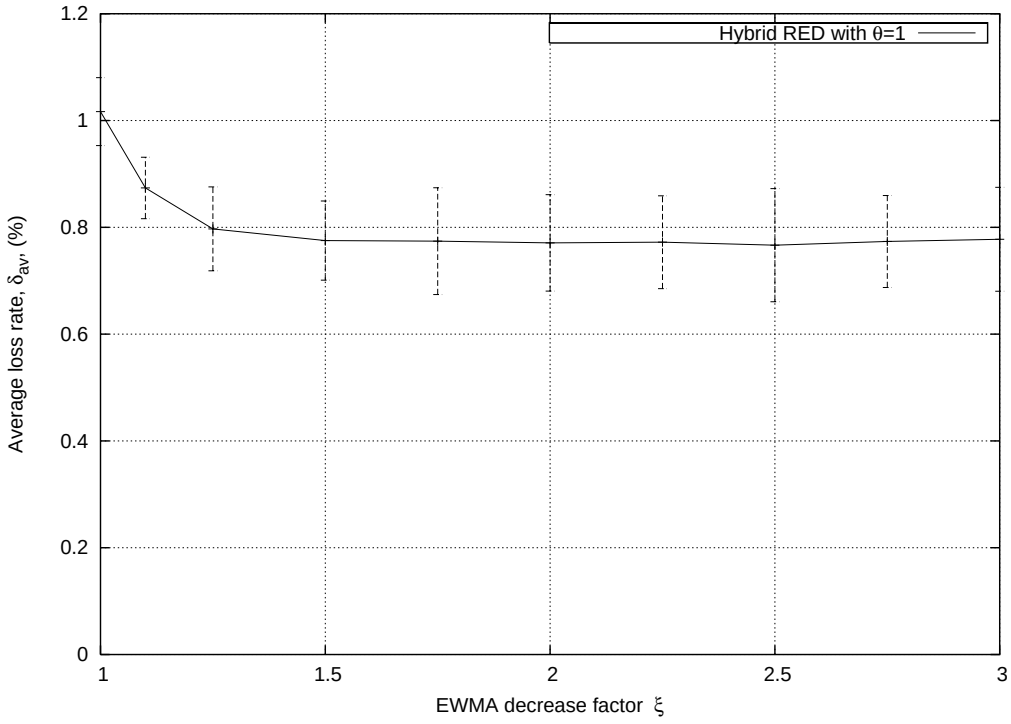

Figure 6.8: Average link loss rate with Hybrid RED for $\theta = 1$ in the simulation topology shown in Figure 6.1.

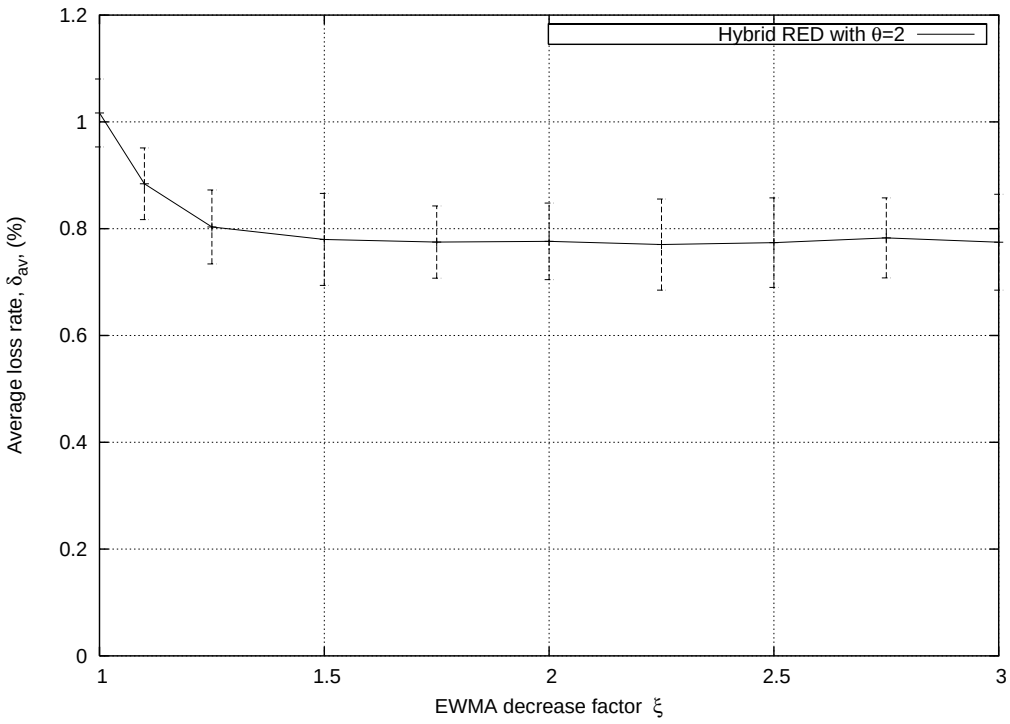

Figure 6.9: Average link loss rate with Hybrid RED for $\theta = 2$ in the simulation topology shown in Figure 6.1.

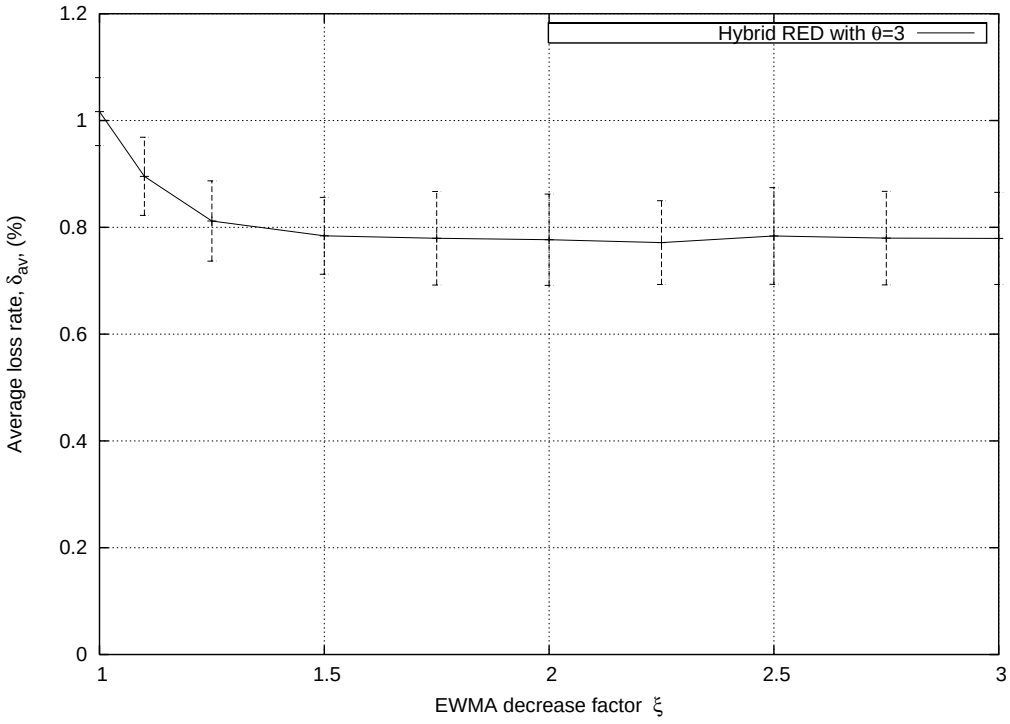

Figure 6.10: Average link loss rate with Hybrid RED for $\theta = 3$ in the simulation topology shown in Figure 6.1.

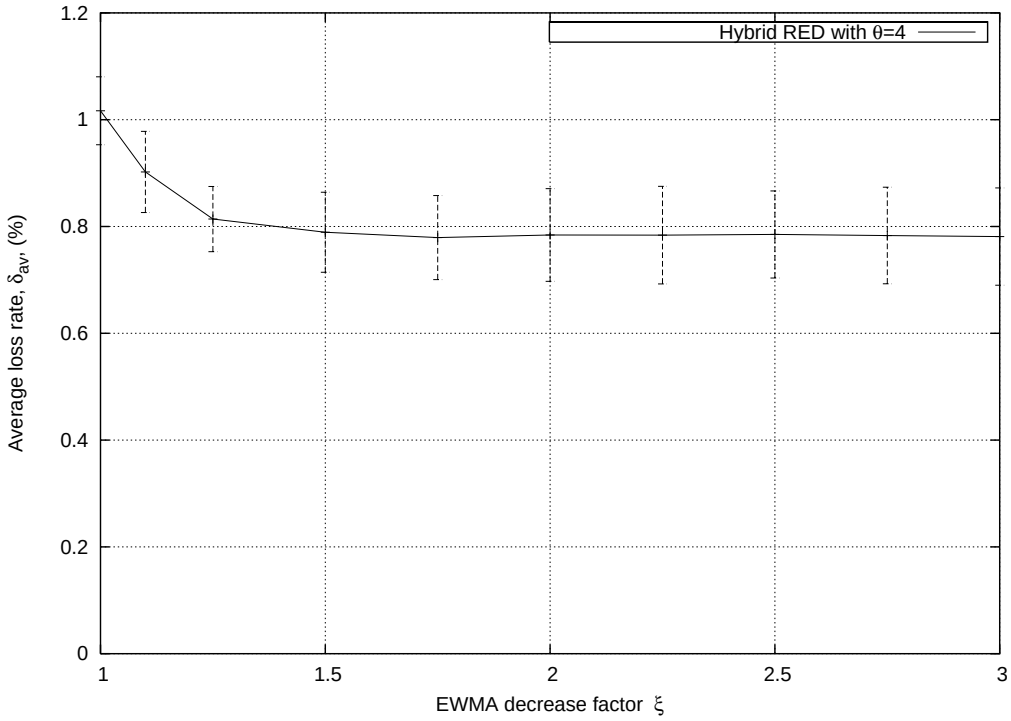

Figure 6.11: Average link loss rate with Hybrid RED for $\theta = 4$ in the simulation topology shown in Figure 6.1.

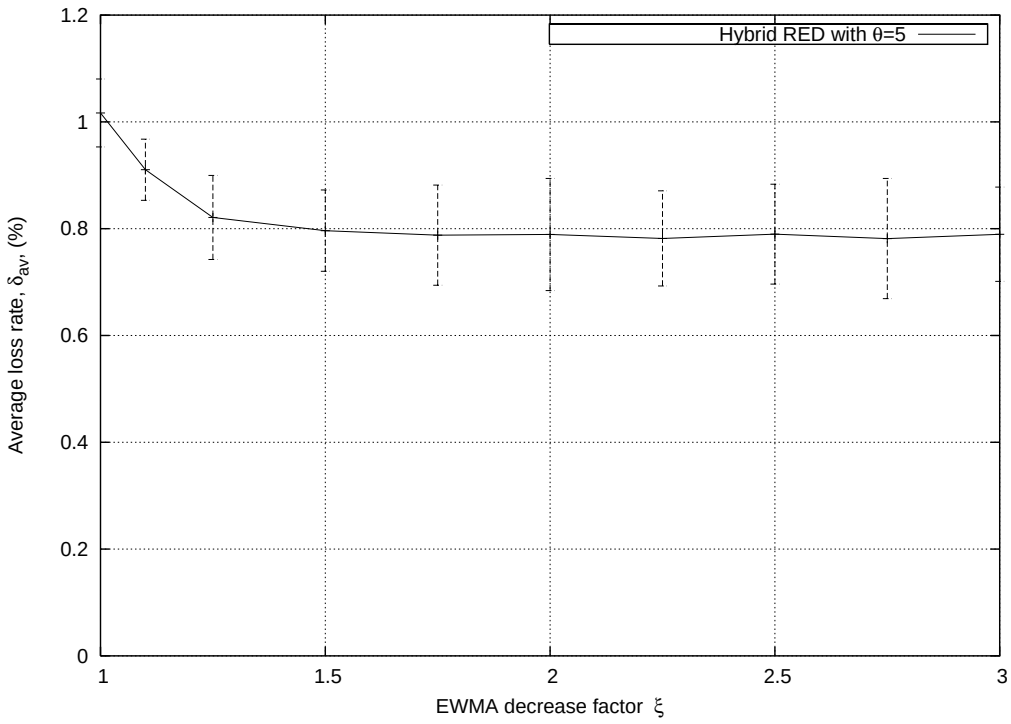

Figure 6.12: Average link loss rate with Hybrid RED for $\theta = 5$ in the simulation topology shown in Figure 6.1.

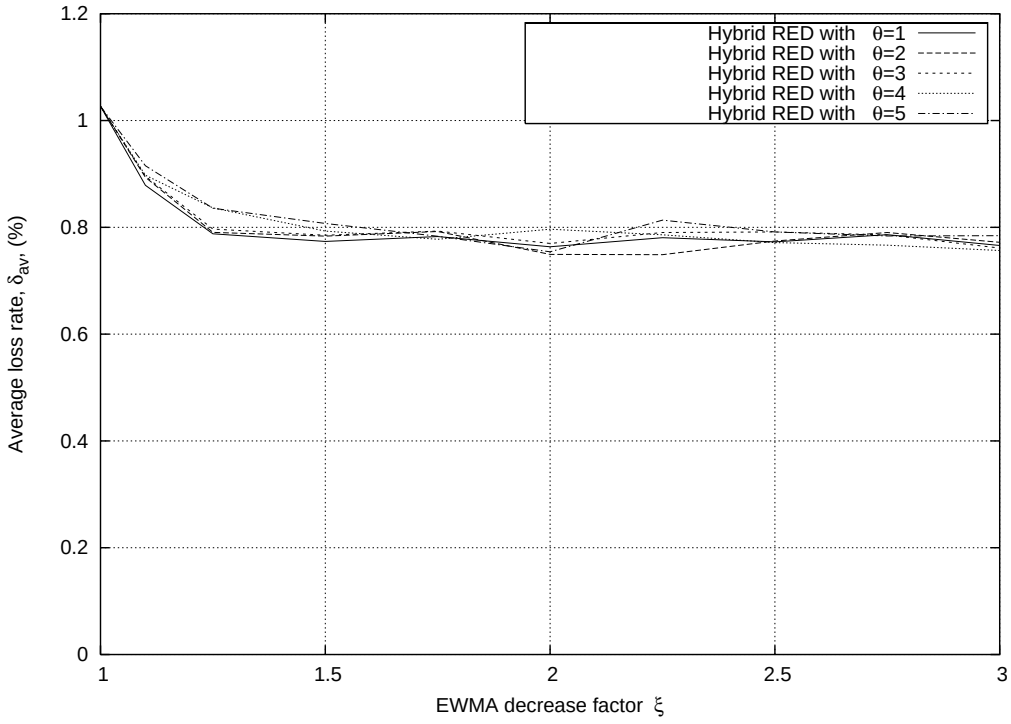

Figure 6.13: Comparison of average link loss rate of Hybrid RED for different values of θ in the simulation topology shown in Figure 6.1.

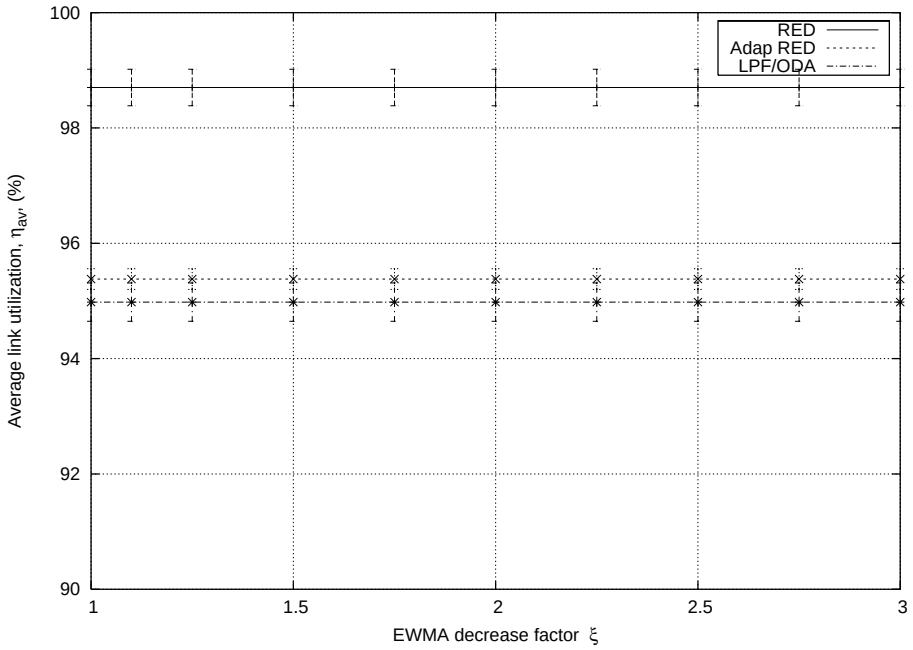

Figure 6.14: Average link utilization with RED, Adaptive RED and LPF/ODA in the simulation topology shown in Figure 6.1.

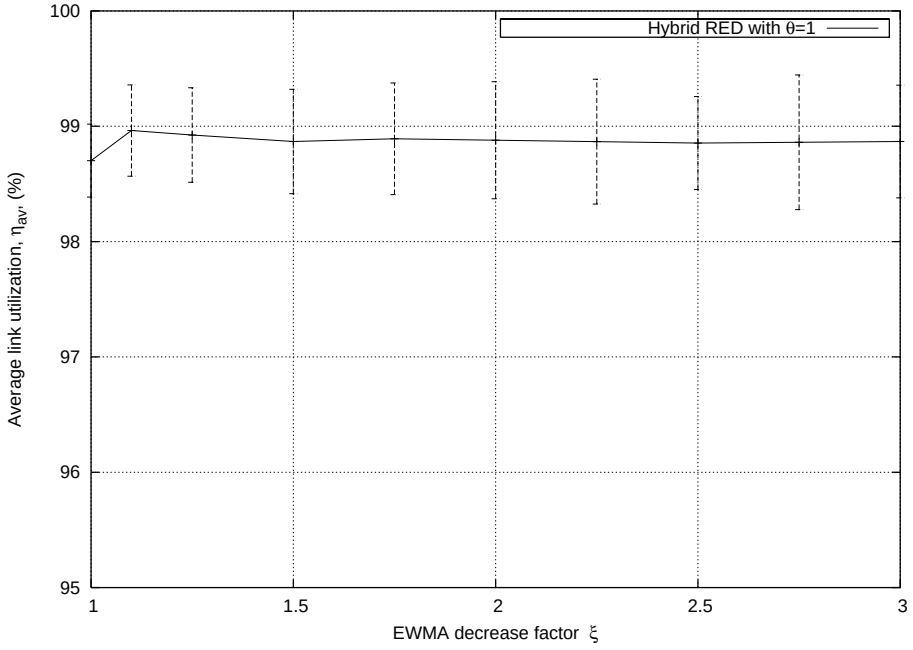

Figure 6.15: Average link utilization with Hybrid RED for $\theta = 1$ in the simulation topology shown in Figure 6.1.

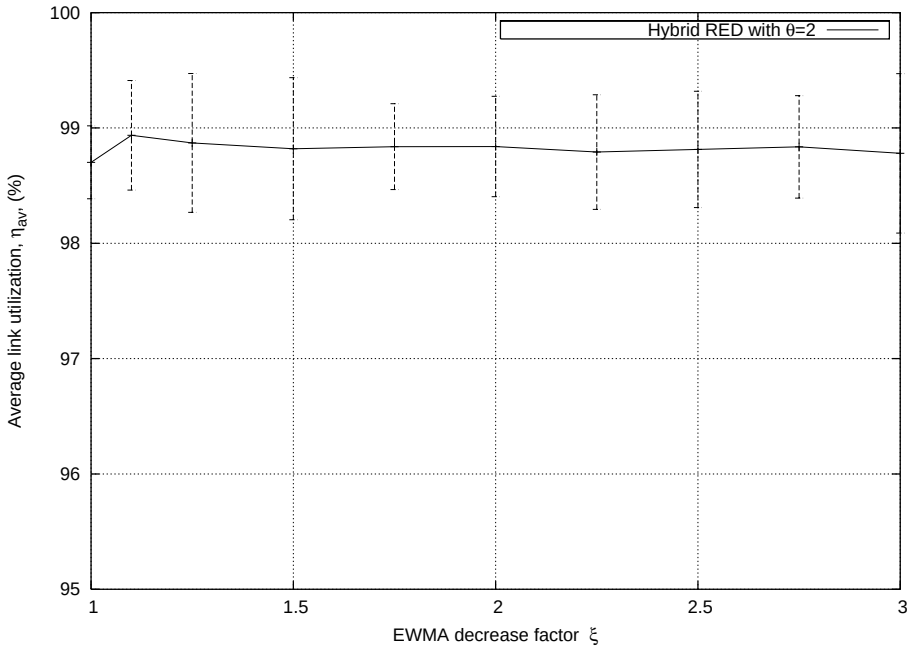

Figure 6.16: Average link utilization with Hybrid RED for $\theta = 2$ in the simulation topology shown in Figure 6.1.

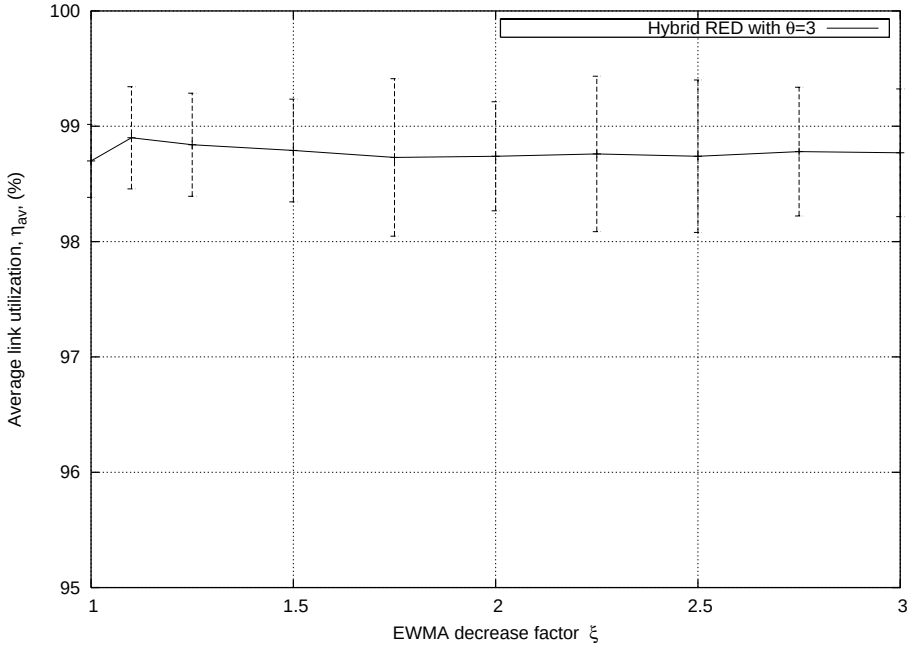

Figure 6.17: Average link utilization with Hybrid RED for $\theta = 3$ in the simulation topology shown in Figure 6.1.

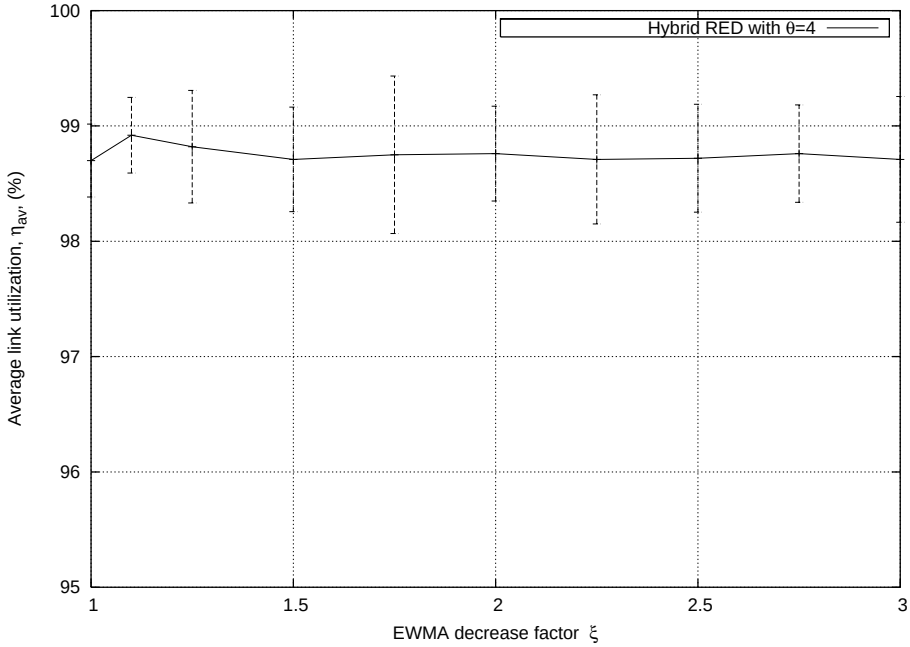

Figure 6.18: Average link utilization with Hybrid RED for $\theta = 4$ in the simulation topology shown in Figure 6.1.

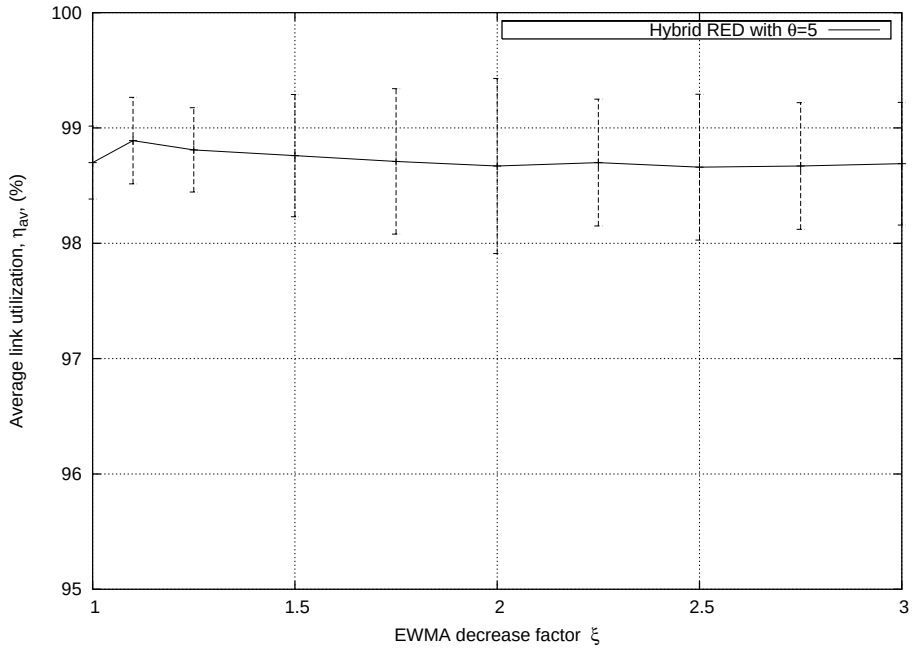

Figure 6.19: Average link utilization with Hybrid RED for $\theta = 5$ in the simulation topology shown in Figure 6.1.

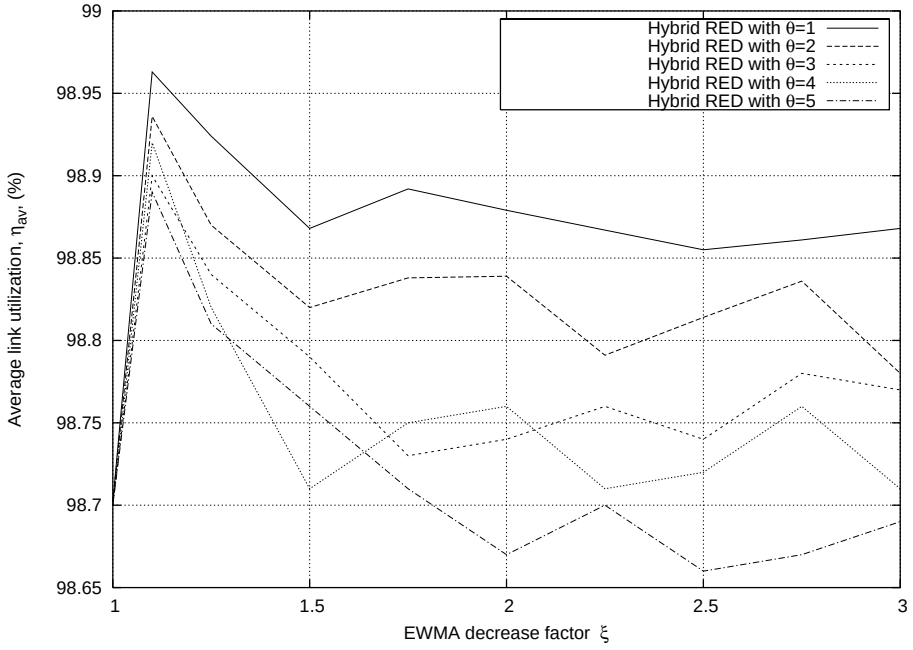

Figure 6.20: Comparison of average link utilization of Hybrid RED for different values of θ in the simulation topology shown in Figure 6.1.

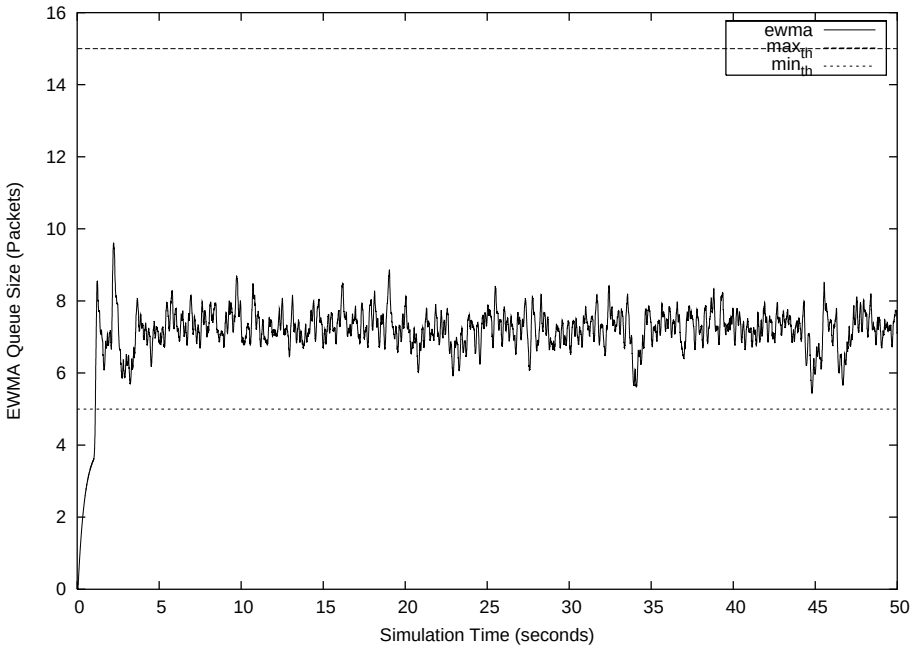

Figure 6.21: EWMA queue size of RED in the simulation topology shown in Figure 6.1.

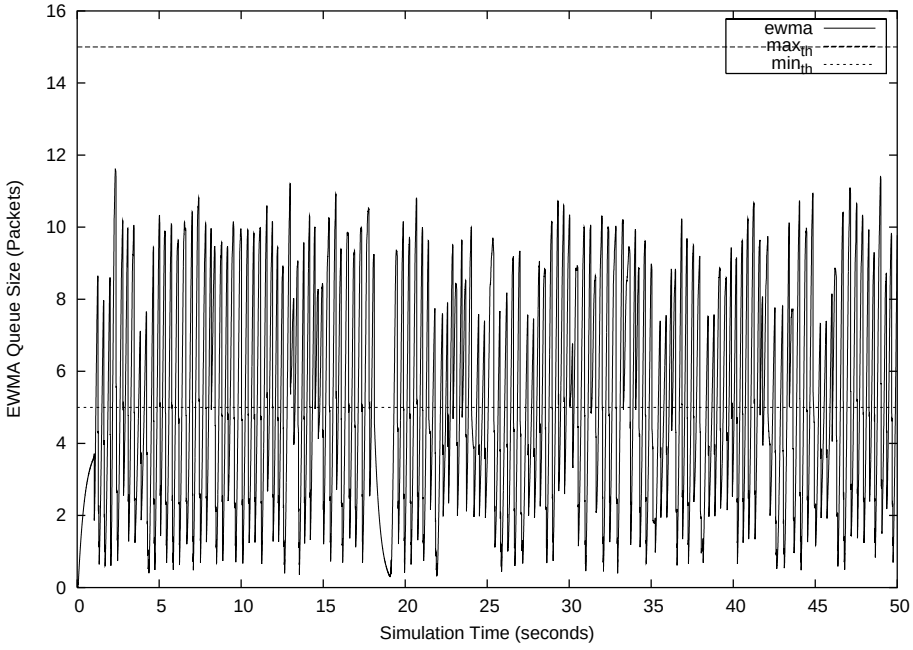

Figure 6.22: EWMA queue size of LPF/ODA in the simulation topology shown in Figure 6.1.

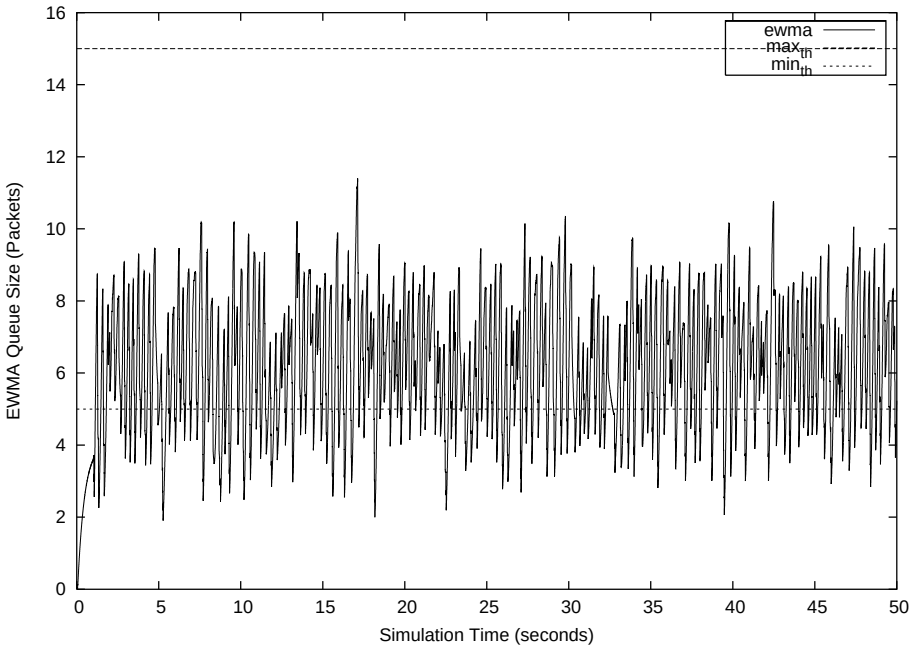

Figure 6.23: EWMA queue size of Hybrid RED with $\{\theta, \xi\} = \{1, 1.25\}$ in the simulation topology shown in Figure 6.1.

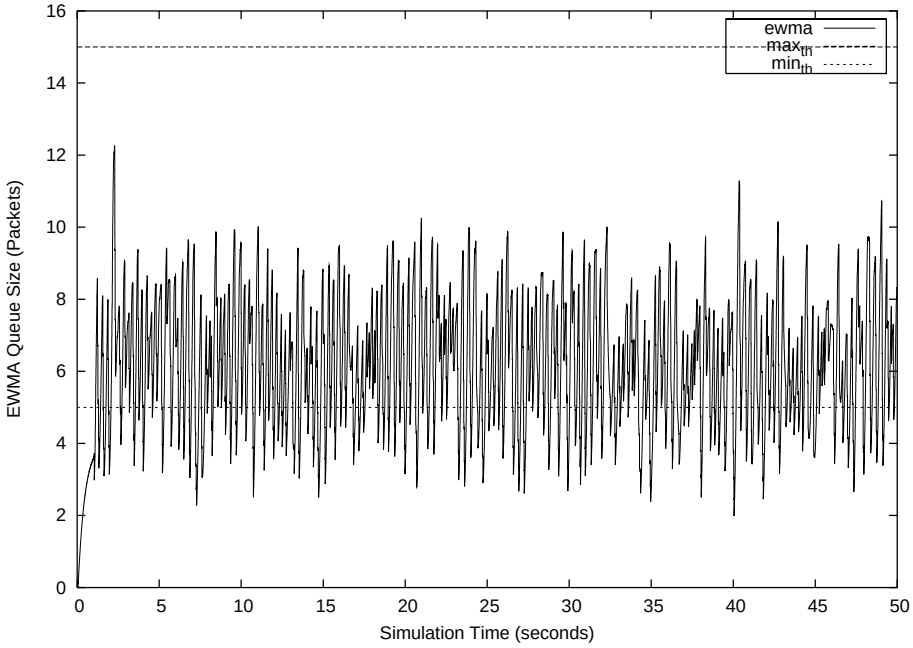

Figure 6.24: EWMA queue size of Hybrid RED with $\{\theta, \xi\} = \{2, 1.25\}$ in the simulation topology shown in Figure 6.1.

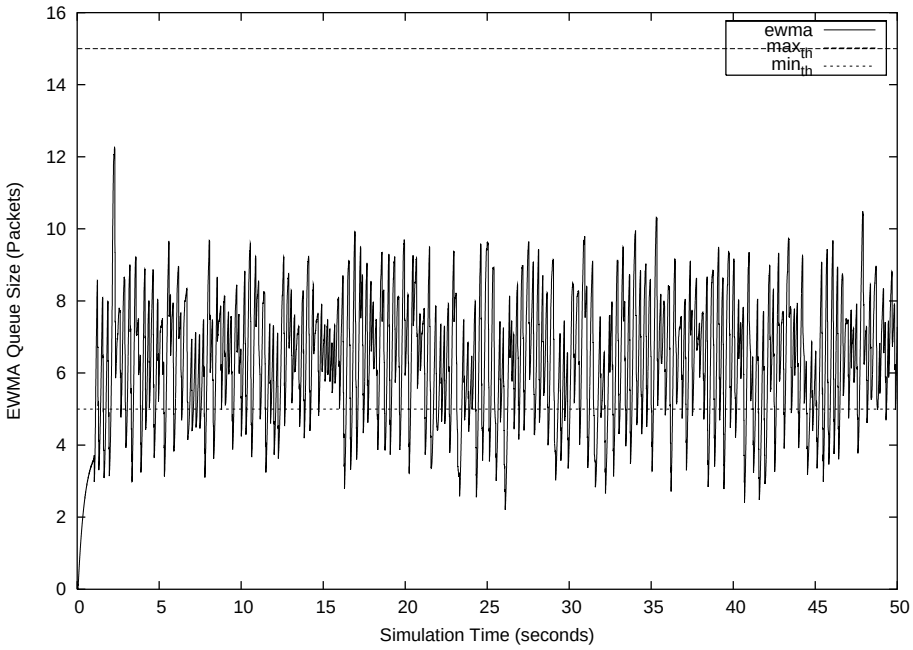

Figure 6.25: EWMA queue size of Hybrid RED with $\{\theta, \xi\} = \{3, 1.25\}$ in the simulation topology shown in Figure 6.1.

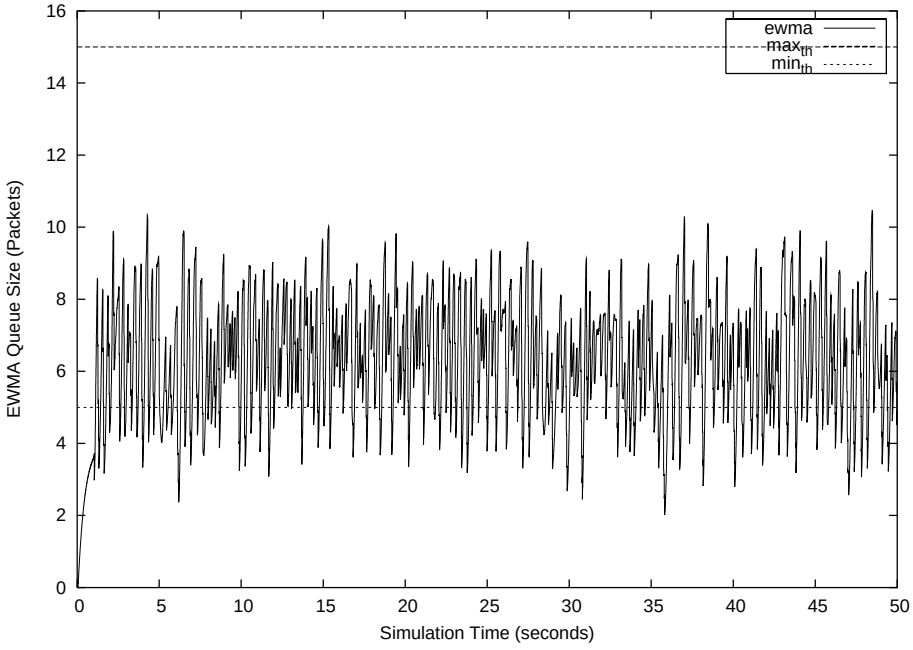

Figure 6.26: EWMA queue size of Hybrid RED with $\{\theta, \xi\} = \{4, 1.25\}$ in the simulation topology shown in Figure 6.1.

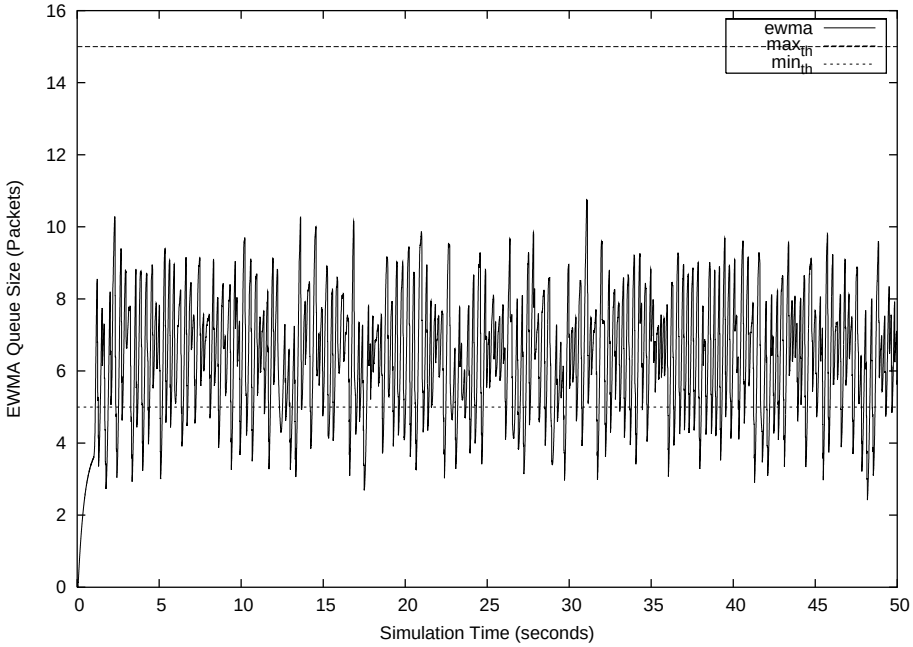

Figure 6.27: EWMA queue size of Hybrid RED with $\{\theta, \xi\} = \{5, 1.25\}$ in the simulation topology shown in Figure 6.1.

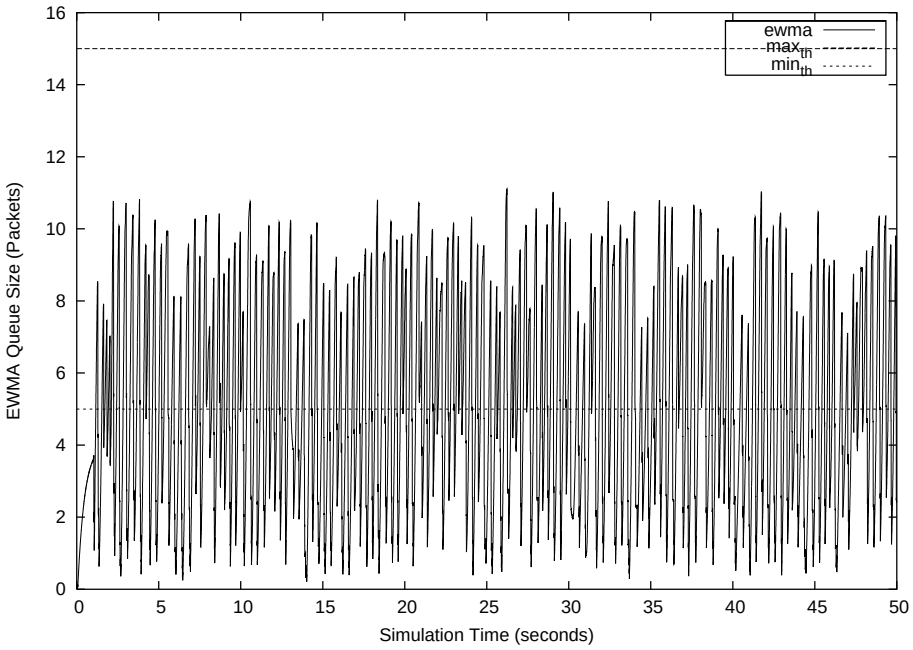

Figure 6.28: EWMA queue size of Hybrid RED with $\{\theta, \xi\} = \{1, 2.0\}$ in the simulation topology shown in Figure 6.1.

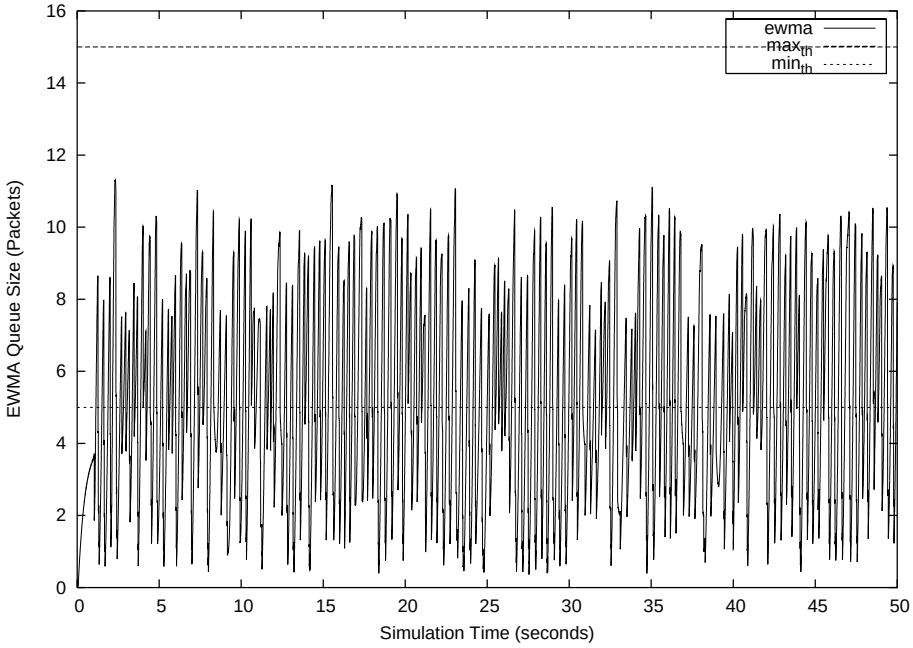

Figure 6.29: EWMA queue size of Hybrid RED with $\{\theta, \xi\} = \{2, 2.0\}$ in the simulation topology shown in Figure 6.1.

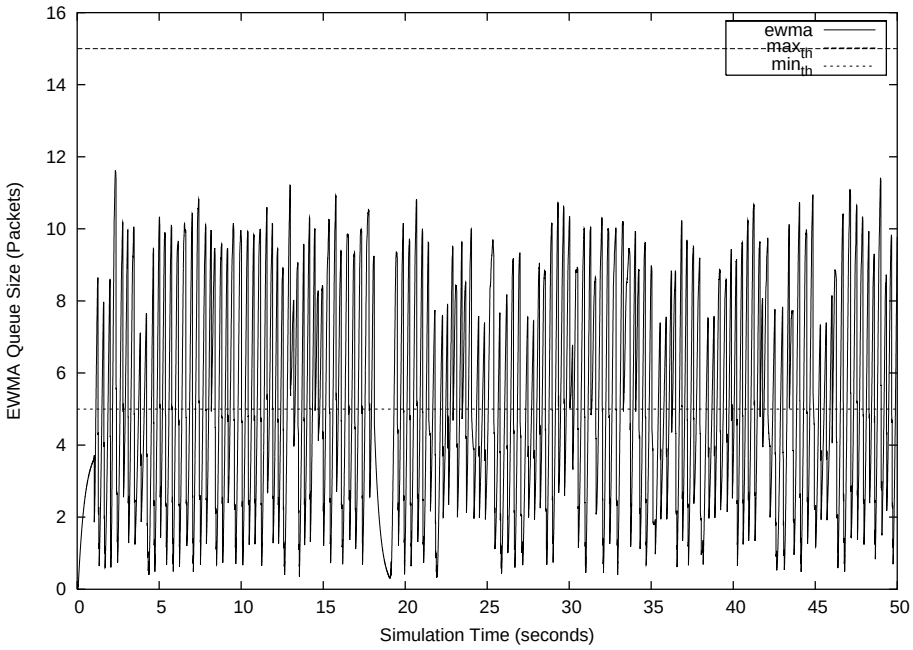

Figure 6.30: EWMA queue size of Hybrid RED with $\{\theta, \xi\} = \{3, 2.0\}$ in the simulation topology shown in Figure 6.1.

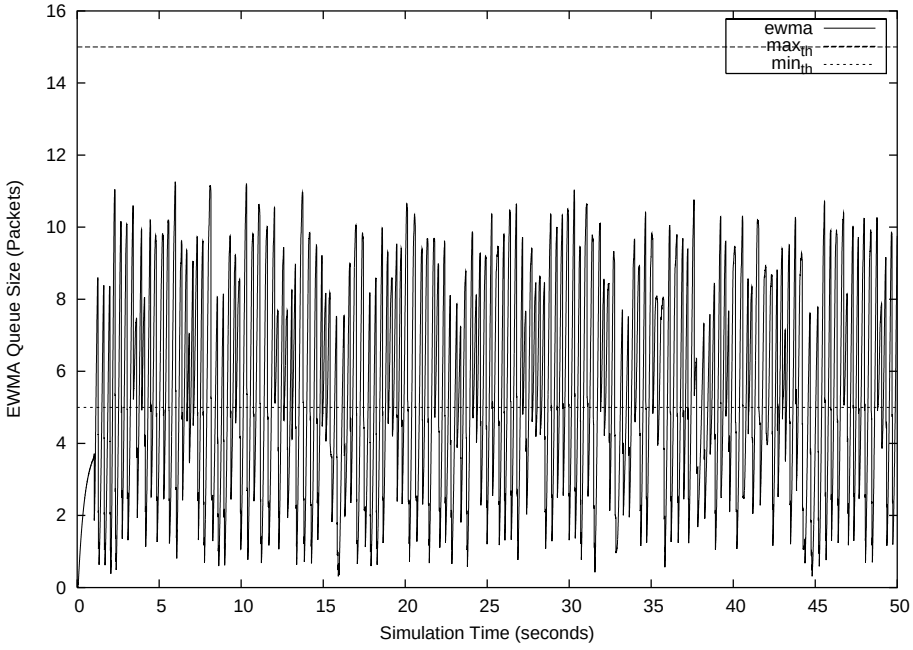

Figure 6.31: EWMA queue size of Hybrid RED with $\{\theta, \xi\} = \{4, 2.0\}$ in the simulation topology shown in Figure 6.1.

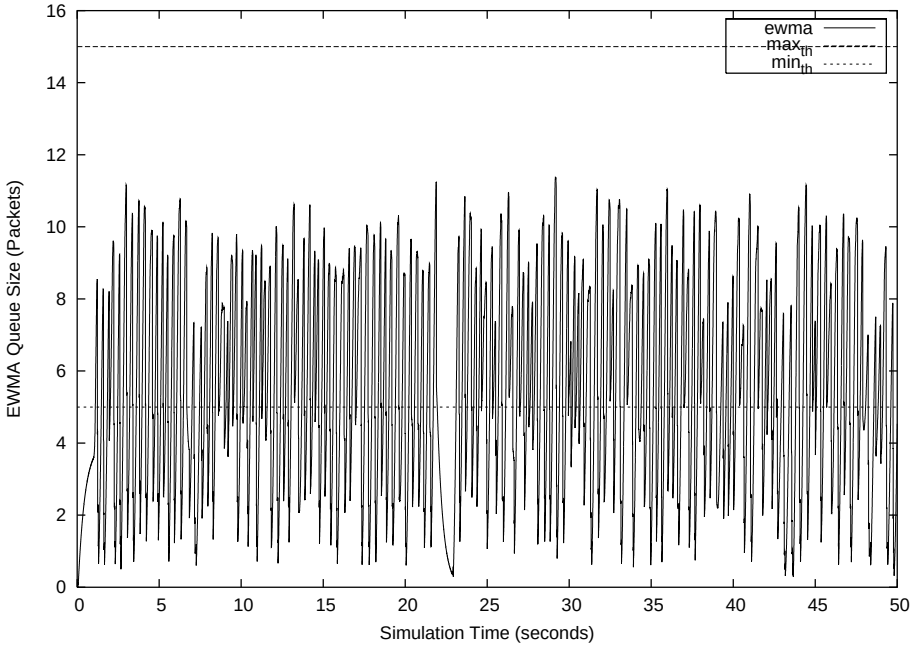

Figure 6.32: EWMA queue size of Hybrid RED with $\{\theta, \xi\} = \{5, 2.0\}$ in the simulation topology shown in Figure 6.1.

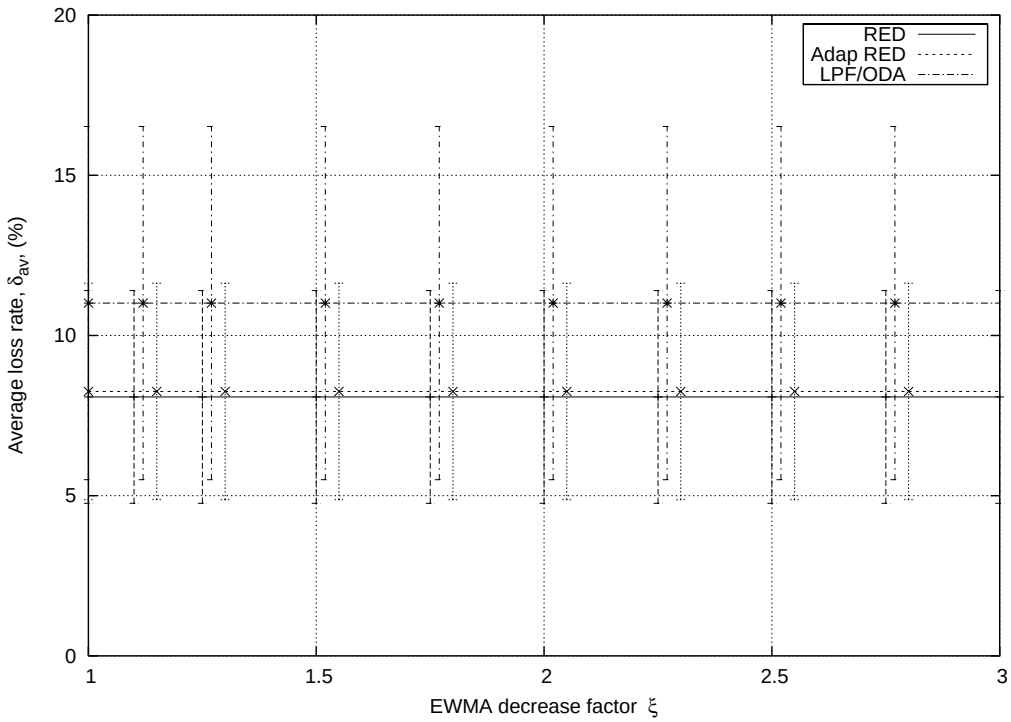

Figure 6.33: Average link loss rate with RED, Adaptive RED and LPF/ODA in the simulation topology shown in Figure 6.2.

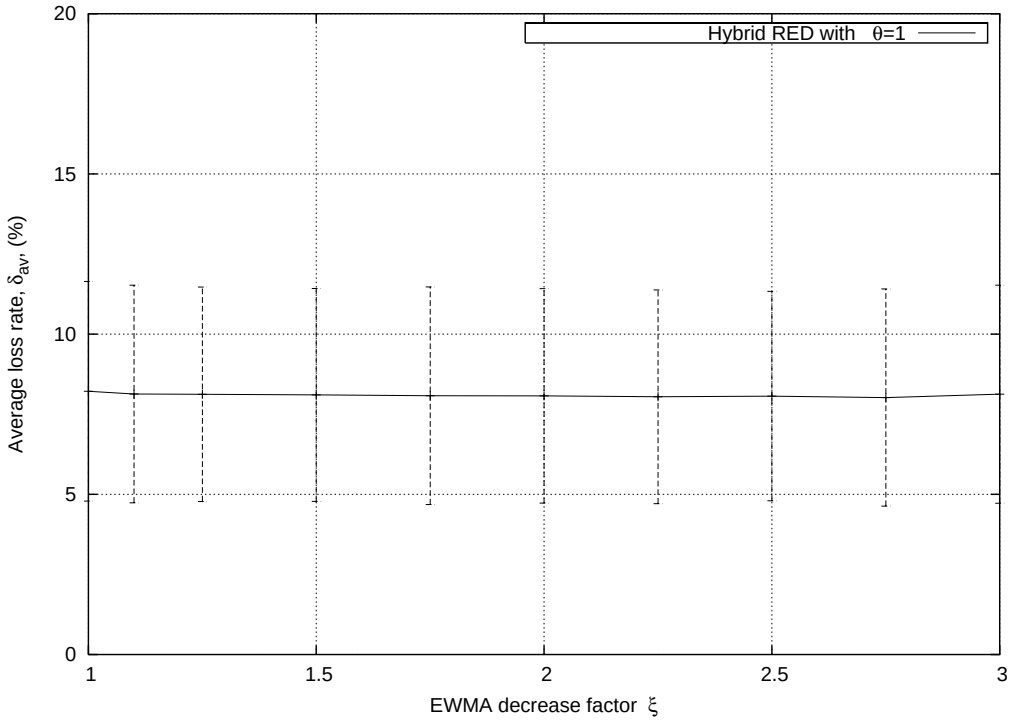

Figure 6.34: Average link loss rate with Hybrid RED for $\theta = 1$, in simulation topology shown in Figure 6.2.

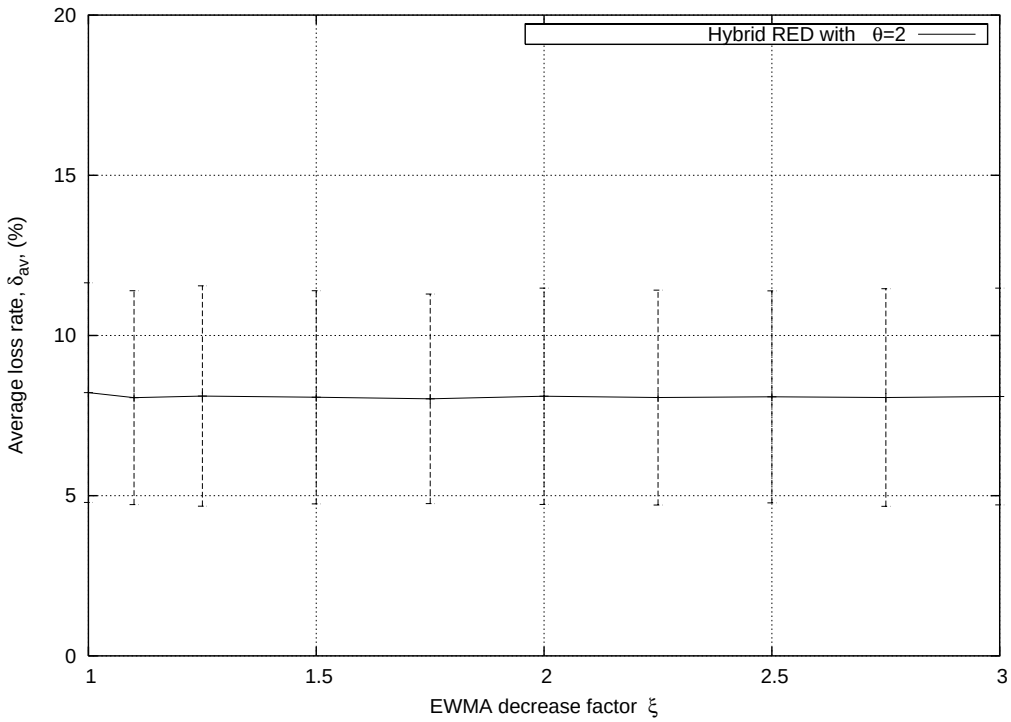

Figure 6.35: Average link loss rate with Hybrid RED for $\theta = 2$, in the simulation topology shown in Figure 6.2.

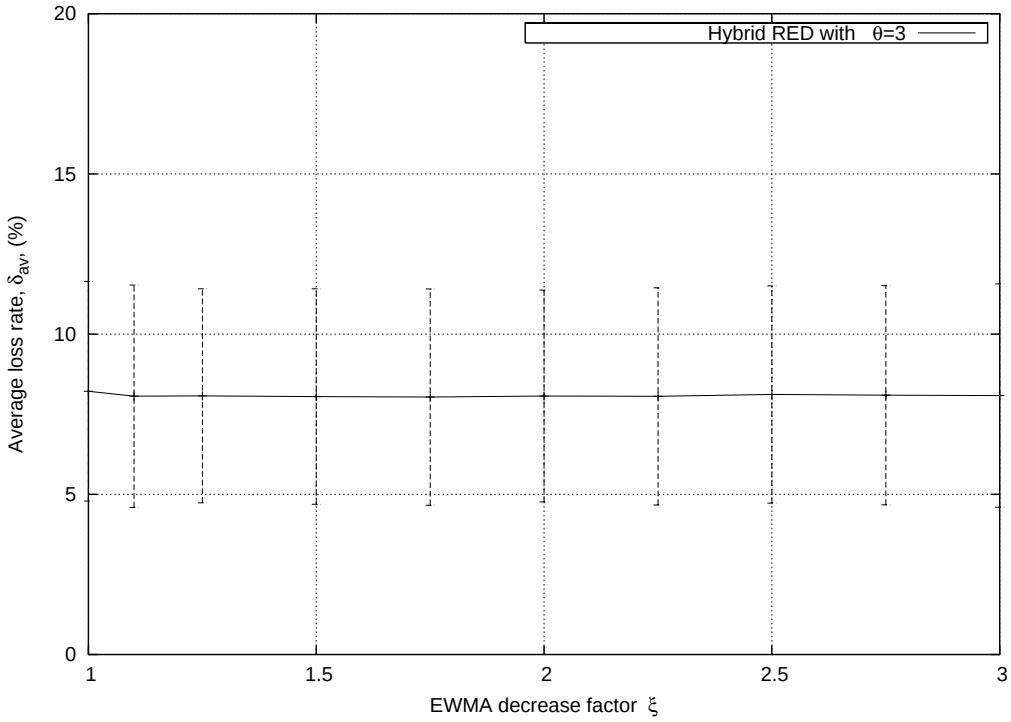

Figure 6.36: Average link loss rate with Hybrid RED for $\theta = 3$, in the simulation topology shown in Figure 6.2.

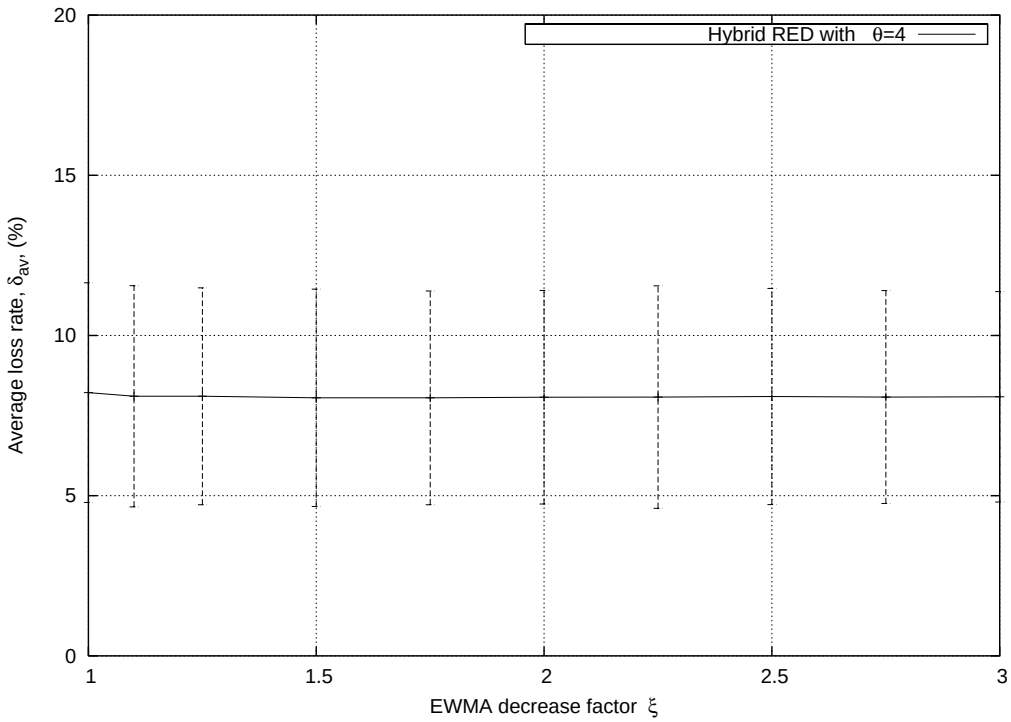

Figure 6.37: Average link loss rate with Hybrid RED for $\theta = 4$, in the simulation topology shown in Figure 6.2.

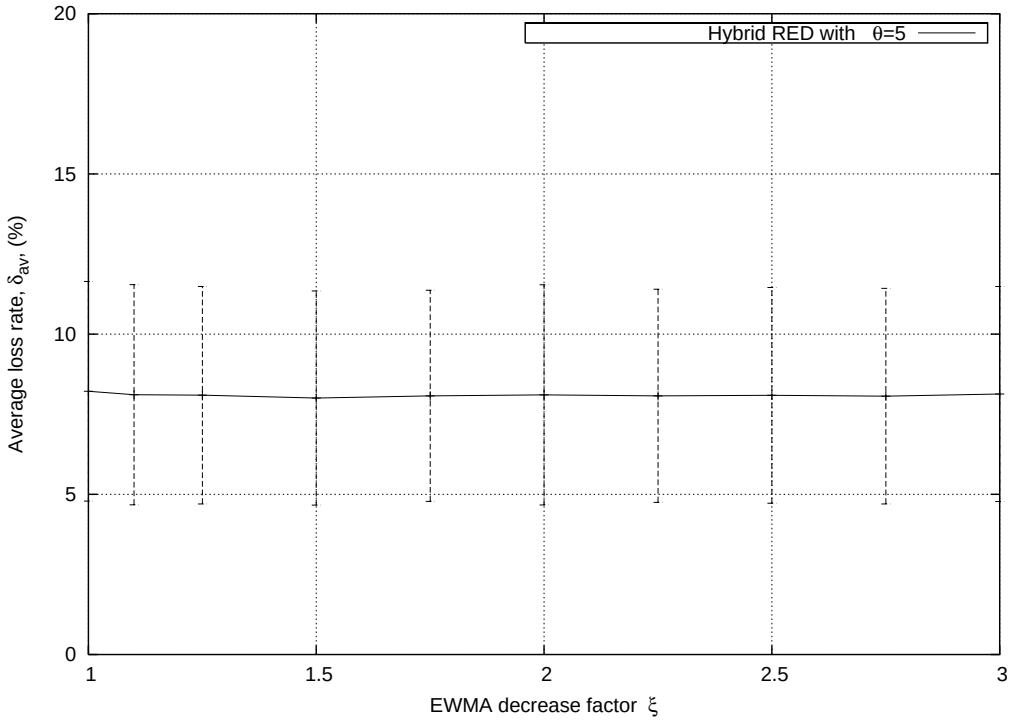

Figure 6.38: Average link loss rate with Hybrid RED for for $\theta = 5$ in the simulation topology shown in Figure 6.2.

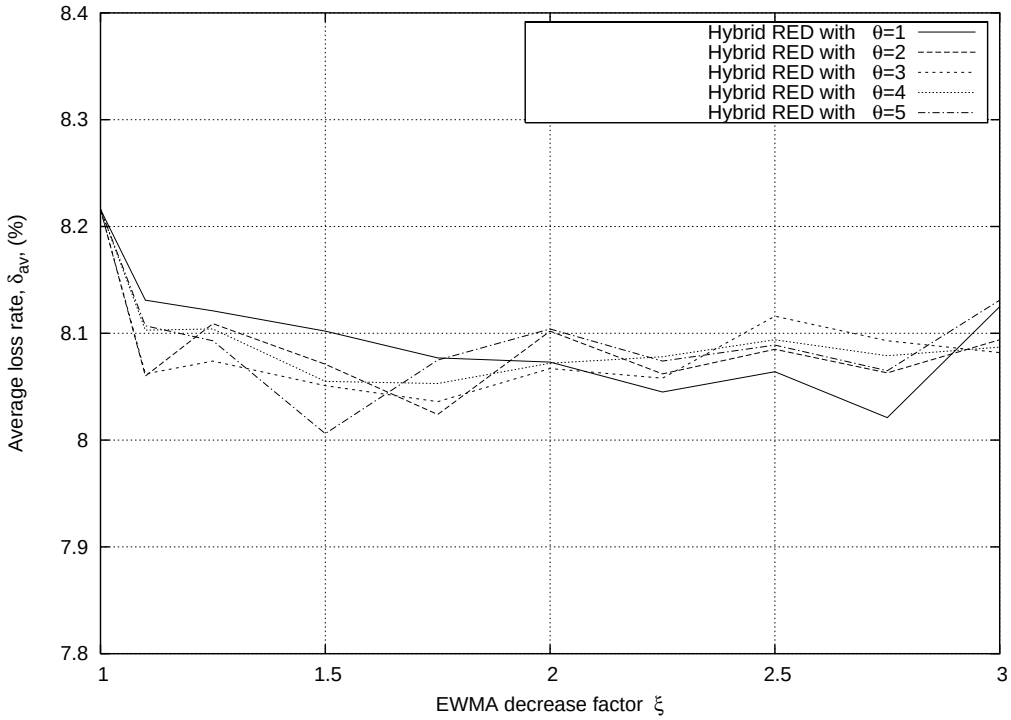

Figure 6.39: Comparison of average link loss rate of Hybrid RED for different values of θ in the simulation topology shown in Figure 6.2.

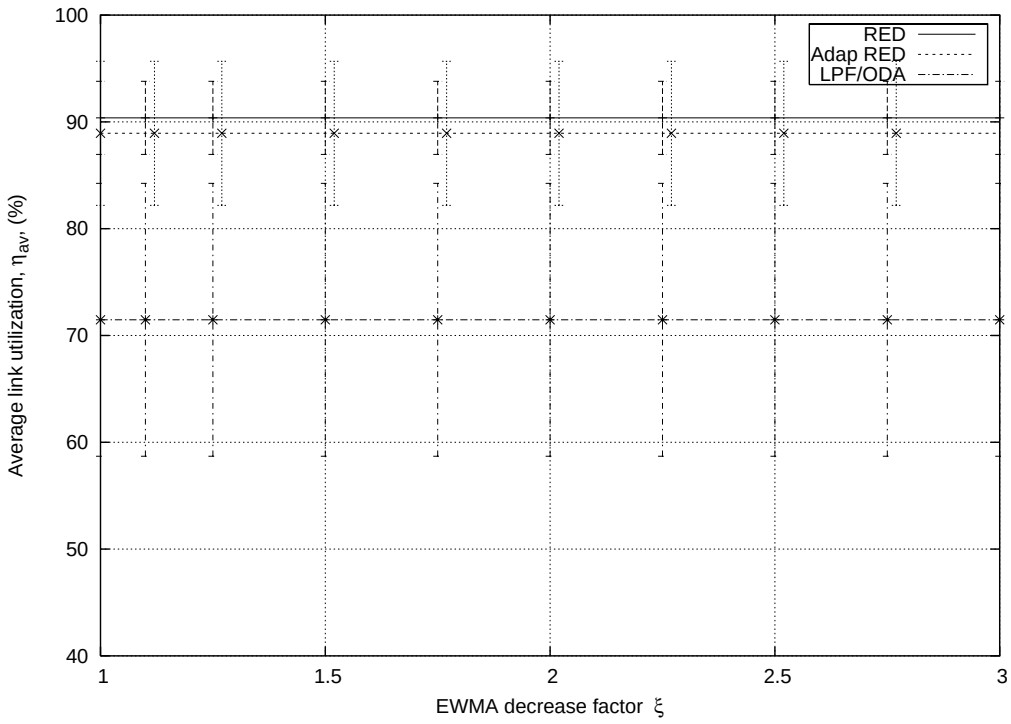

Figure 6.40: Average link utilization with RED, Adaptive RED and LPF/ODA in the simulation topology shown in Figure 6.2.

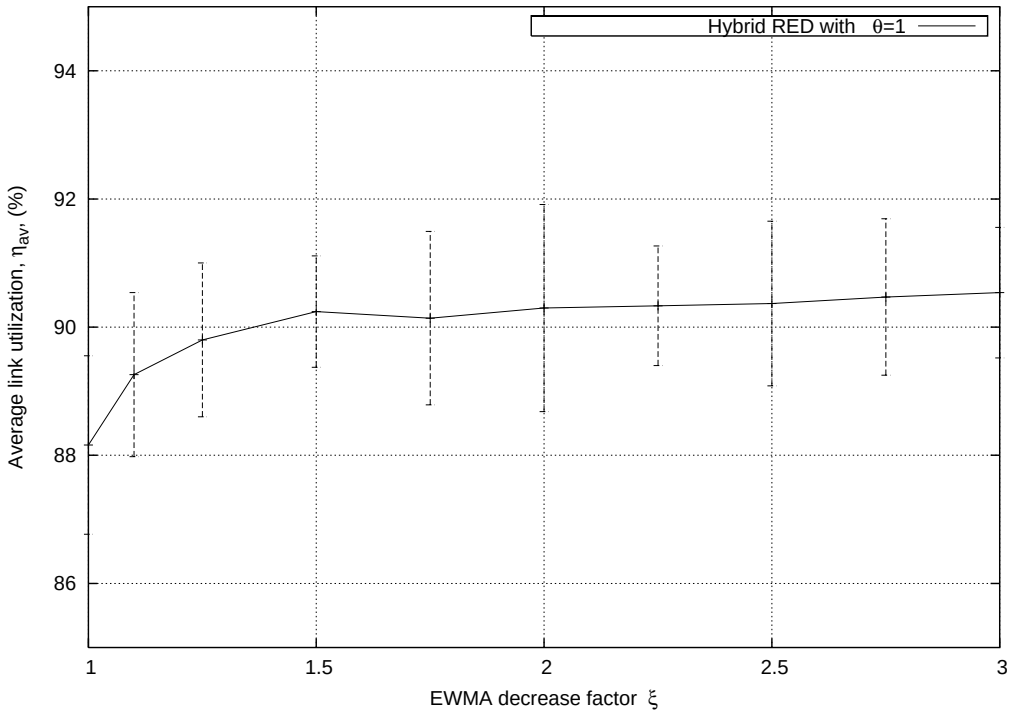

Figure 6.41: Average link utilization with Hybrid RED for $\theta = 1$, in the simulation topology shown in Figure 6.2.

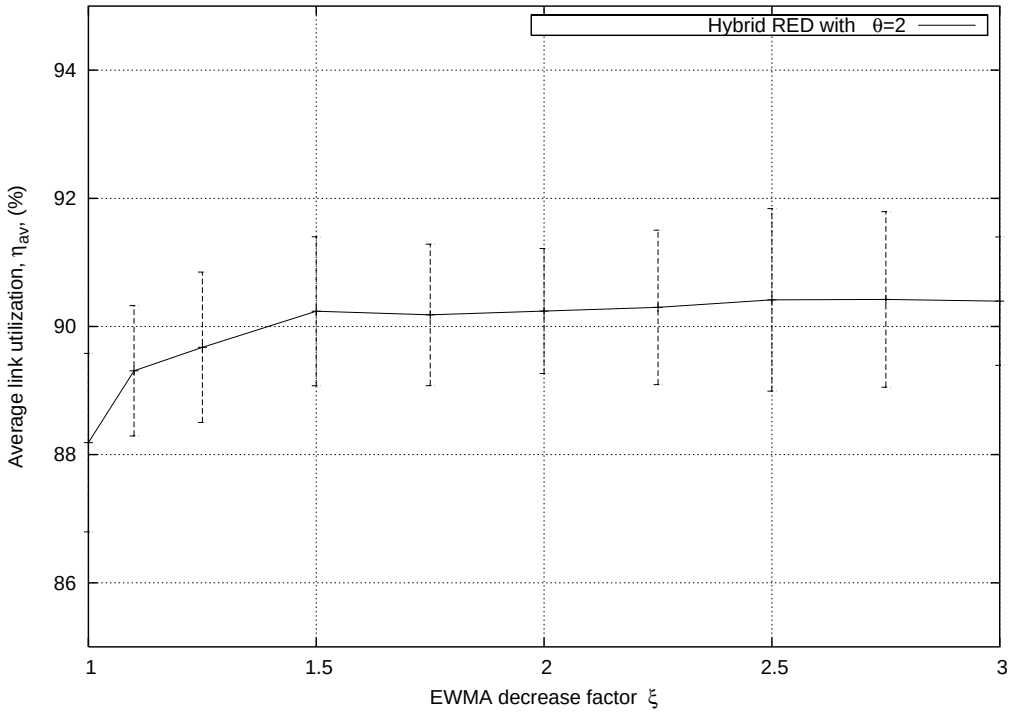

Figure 6.42: Average link utilization with Hybrid RED for $\theta = 2$, in the simulation topology shown in Figure 6.2.

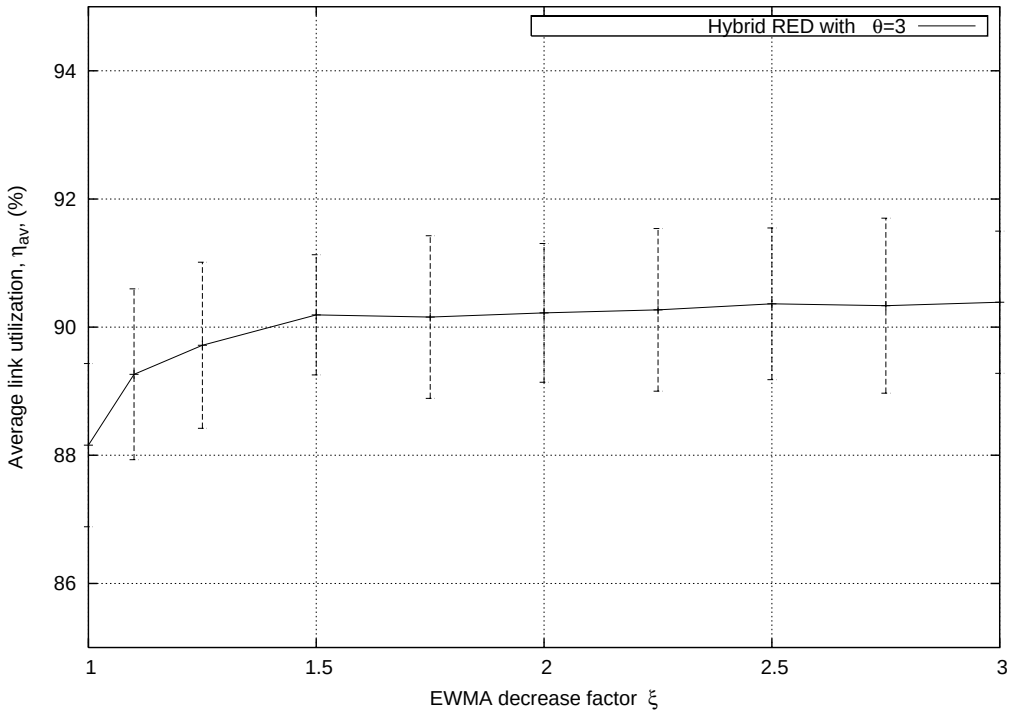

Figure 6.43: Average link utilization with Hybrid RED for $\theta = 3$, in the simulation topology shown in Figure 6.2.

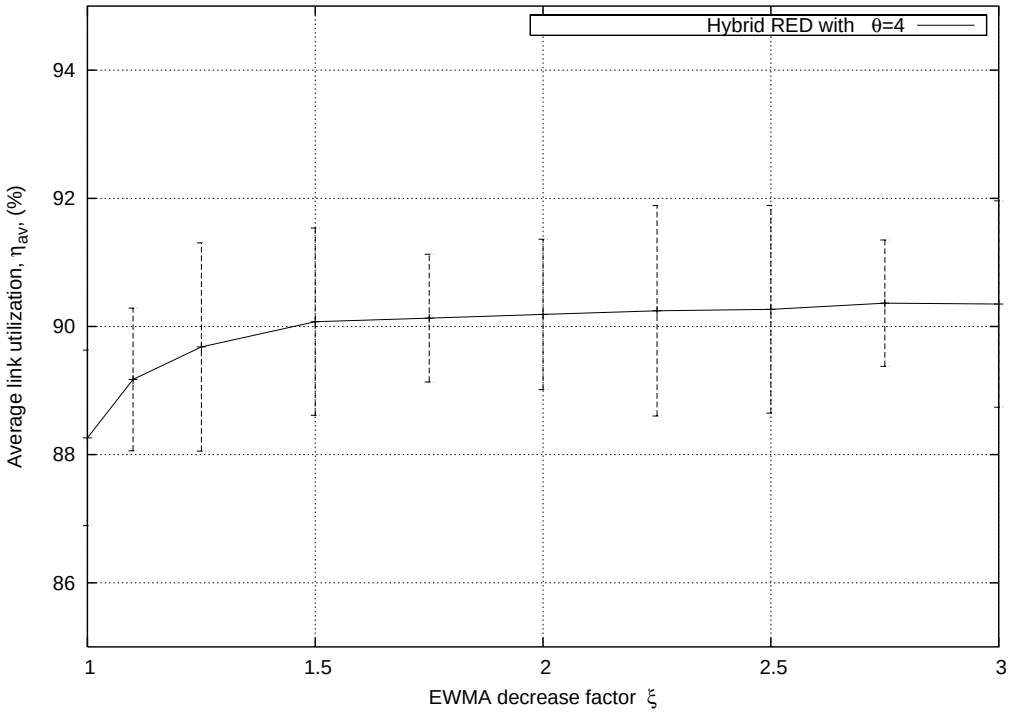

Figure 6.44: Average link utilization with Hybrid RED for $\theta = 4$, in the simulation topology shown in Figure 6.2.

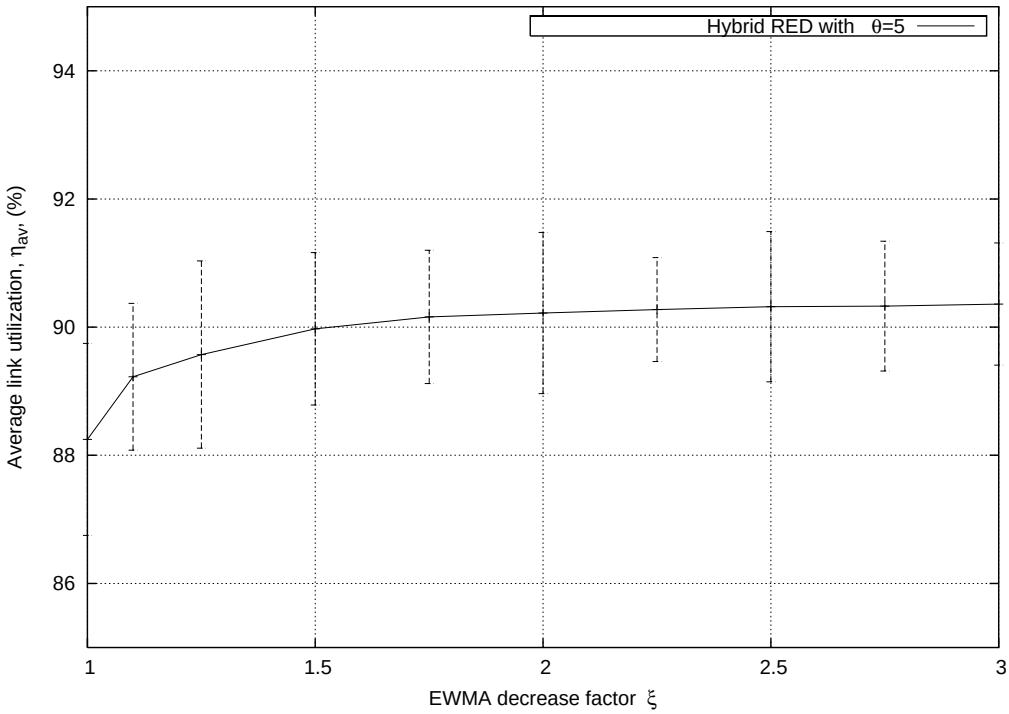

Figure 6.45: Average link utilization with Hybrid RED for $\theta = 5$, in the simulation topology shown in Figure 6.2.

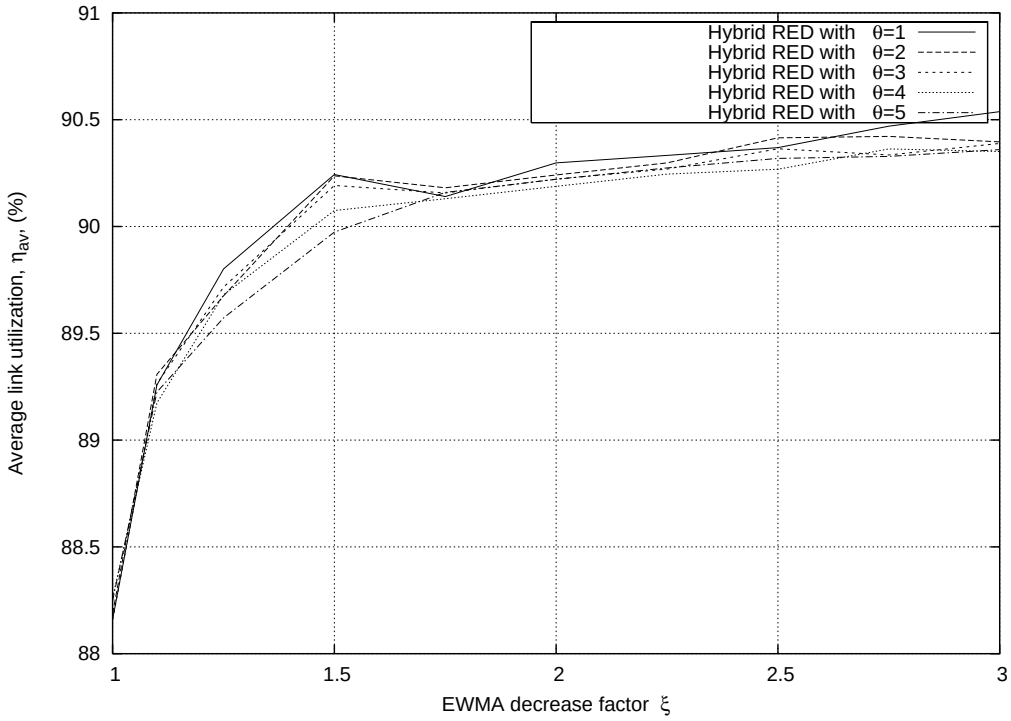

Figure 6.46: Comparison of average link utilization with Hybrid RED algorithm in the simulation topology shown in Figure 6.2.

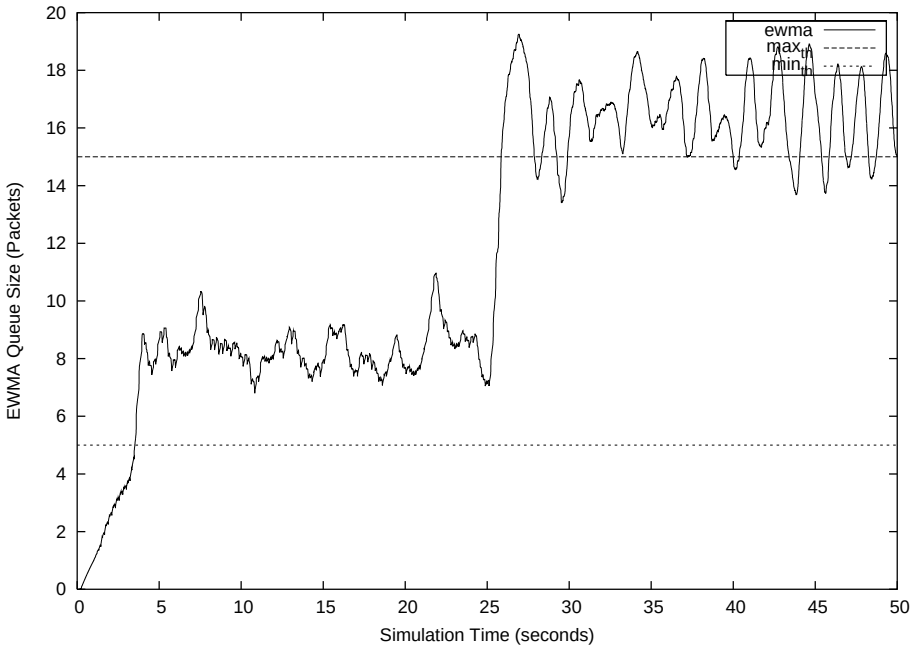

Figure 6.47: EWMA queue size of RED for simulation topology shown in Figure 6.2.

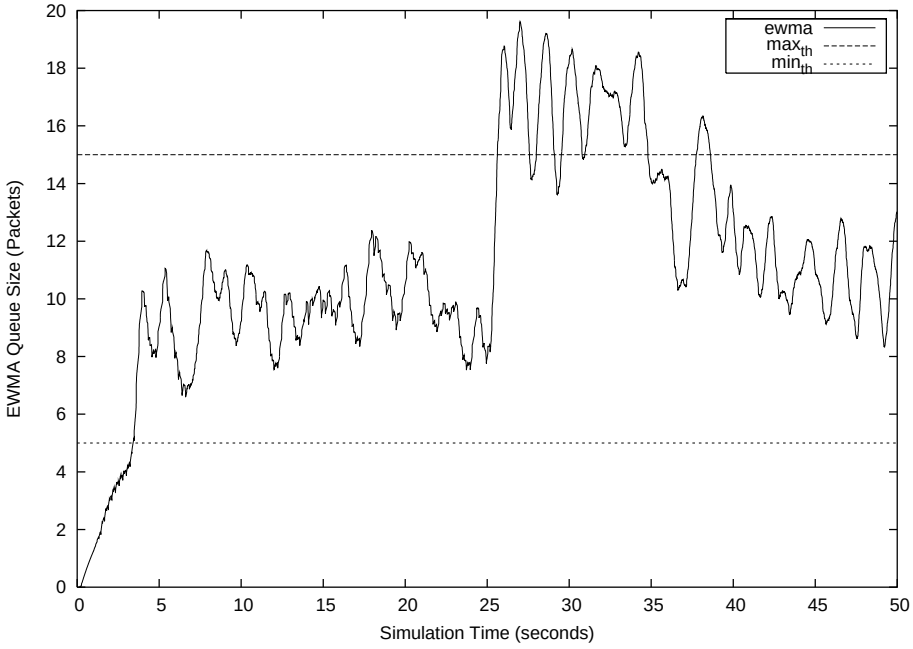

Figure 6.48: EWMA queue size of Adaptive RED for simulation topology shown in Figure 6.2.

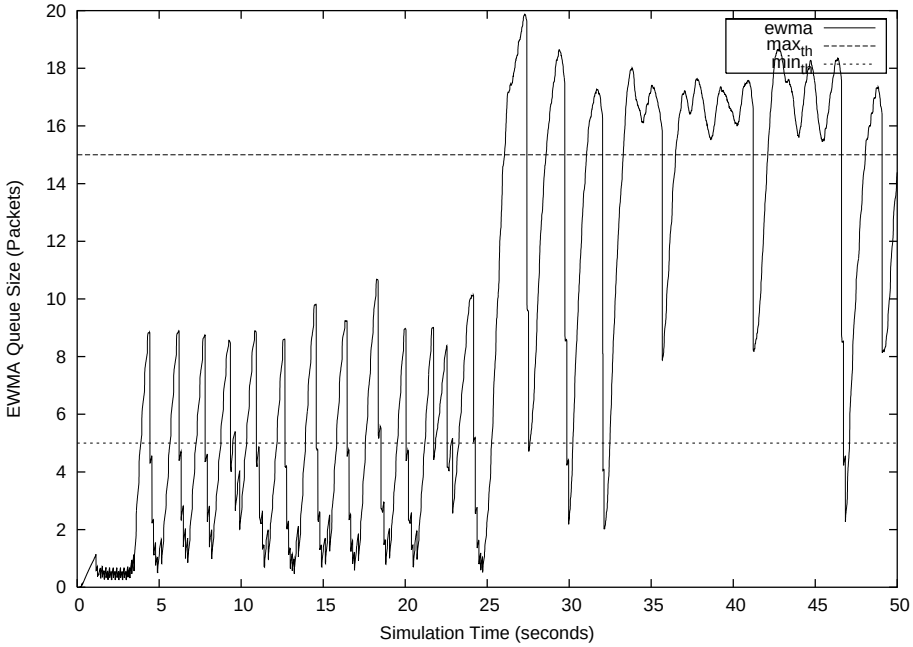

Figure 6.49: EWMA queue size of LPF/ODA for simulation topology shown in Figure 6.2.

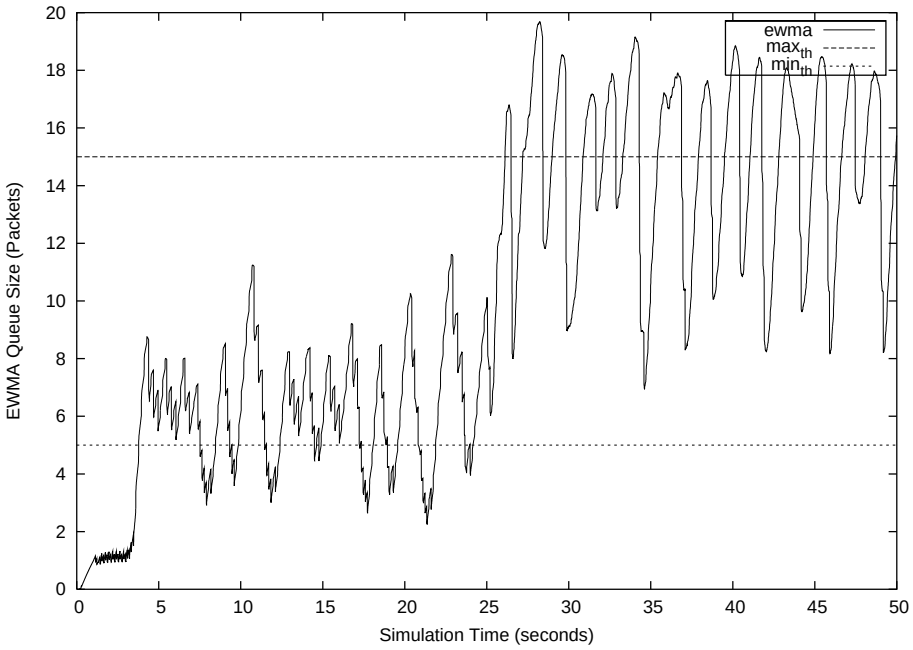

Figure 6.50: EWMA queue size of Hybrid RED with $\{\theta, \xi\} = \{1, 1.25\}$ in simulation topology shown in Figure 6.2.

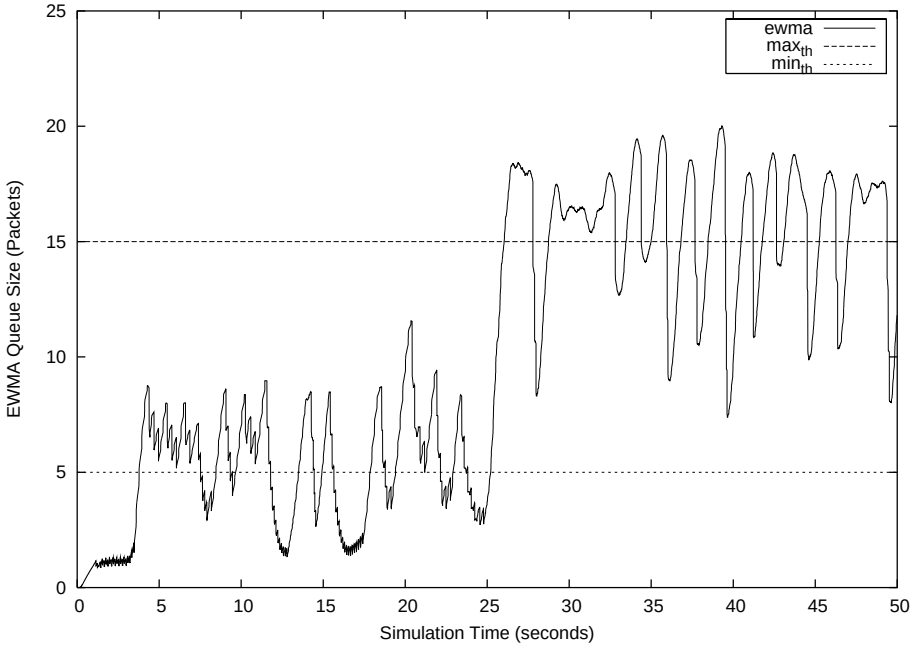

Figure 6.51: EWMA queue size of Hybrid RED with $\{\theta, \xi\} = \{2, 1.25\}$ in simulation topology shown in Figure 6.2.

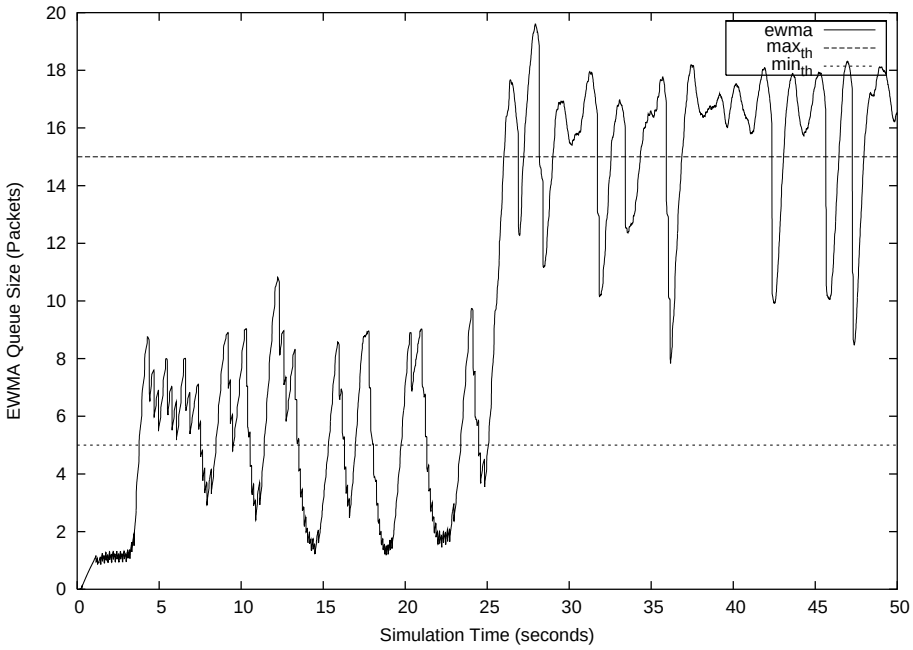

Figure 6.52: EWMA queue size of Hybrid RED with $\{\theta, \xi\} = \{3, 1.25\}$ in simulation topology shown in Figure 6.2.

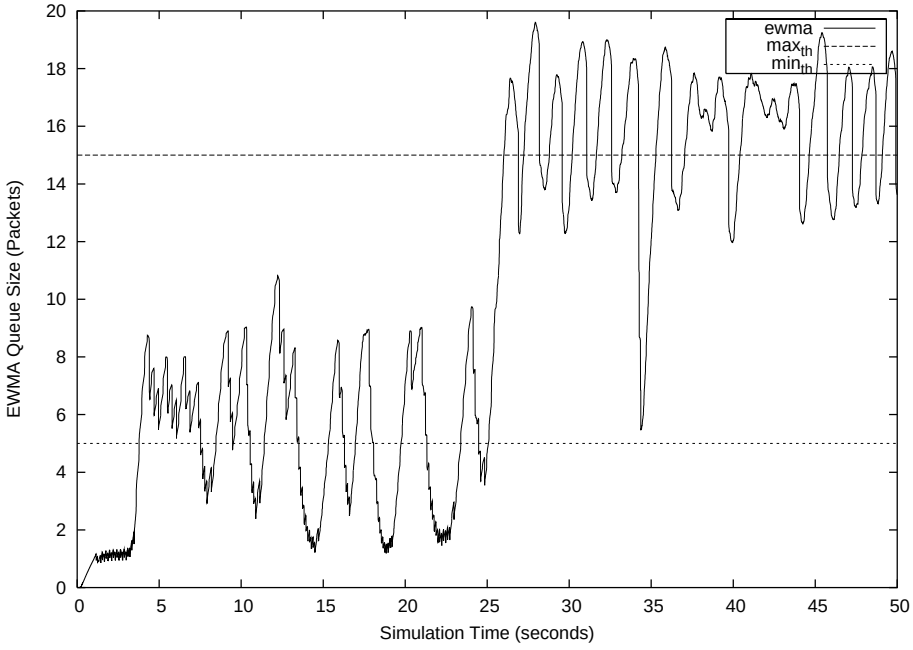

Figure 6.53: EWMA queue size of Hybrid RED with $\{\theta, \xi\} = \{4, 1.25\}$ in simulation topology shown in Figure 6.2.

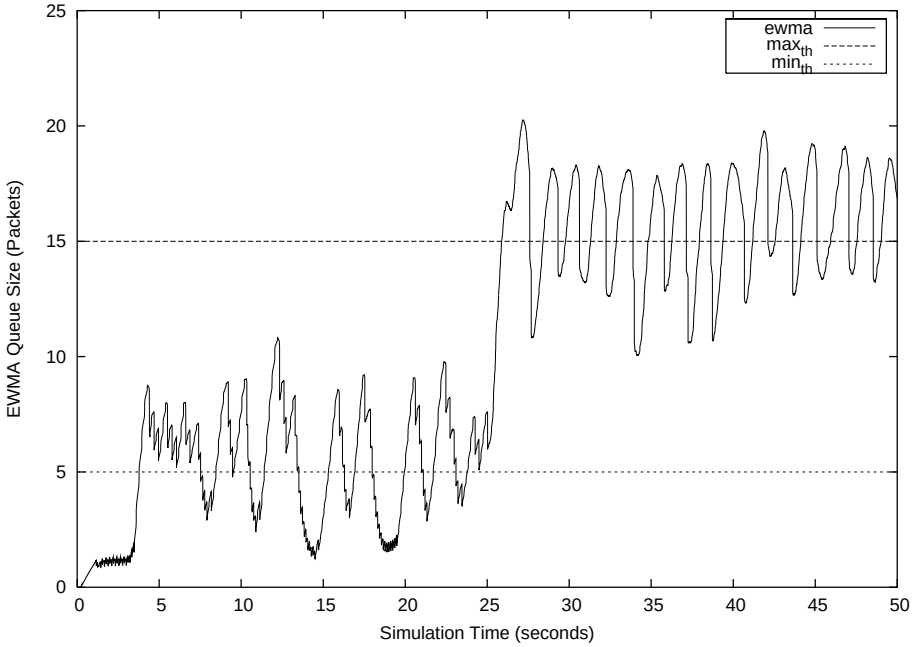

Figure 6.54: EWMA queue size of Hybrid RED with $\{\theta, \xi\} = \{5, 1.25\}$ in simulation topology shown in Figure 6.2.

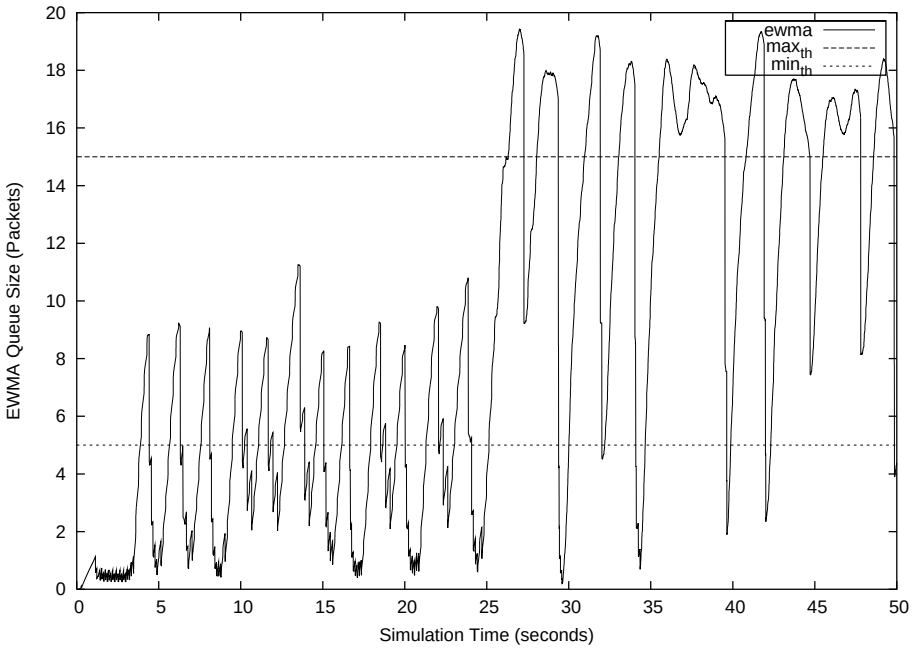

Figure 6.55: EWMA queue size of Hybrid RED with $\{\theta, \xi\} = \{1, 2.0\}$ in simulation topology shown in Figure 6.2.

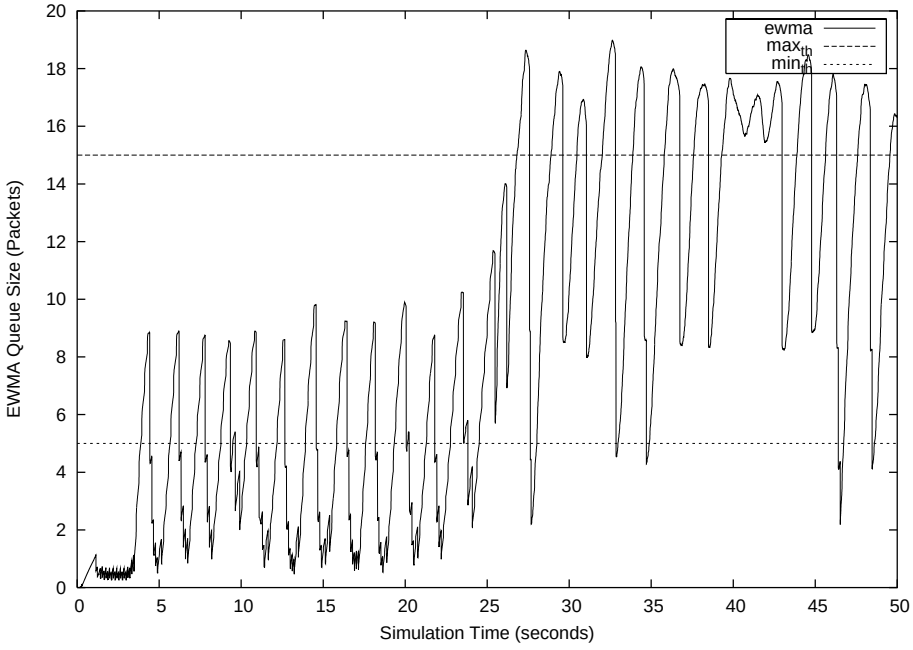

Figure 6.56: EWMA queue size of Hybrid RED with $\{\theta, \xi\} = \{2, 2.0\}$ in simulation topology shown in Figure 6.2.

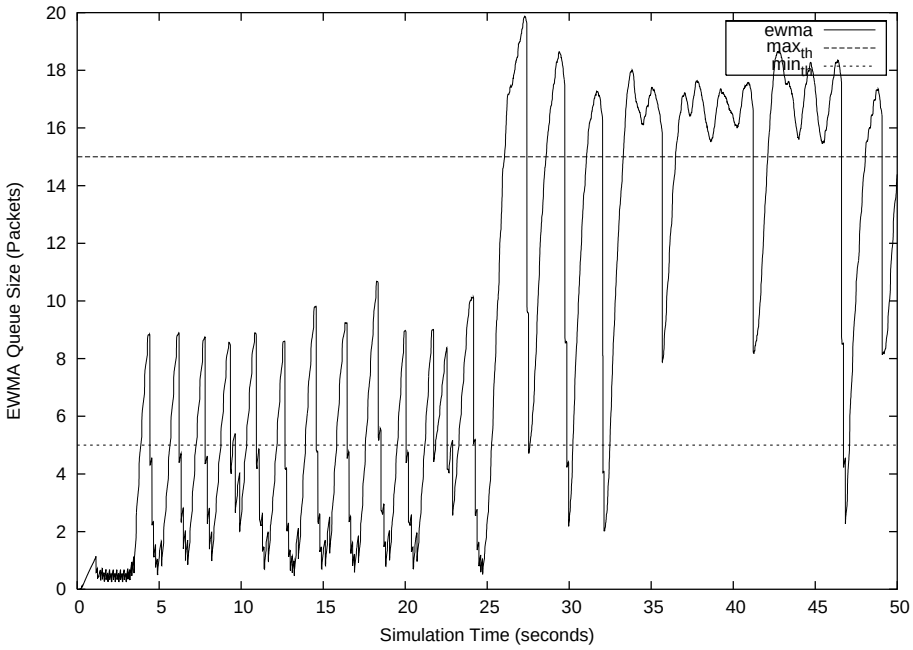

Figure 6.57: EWMA queue size of Hybrid RED with $\{\theta, \xi\} = \{3, 2.0\}$ in simulation topology shown in Figure 6.2.

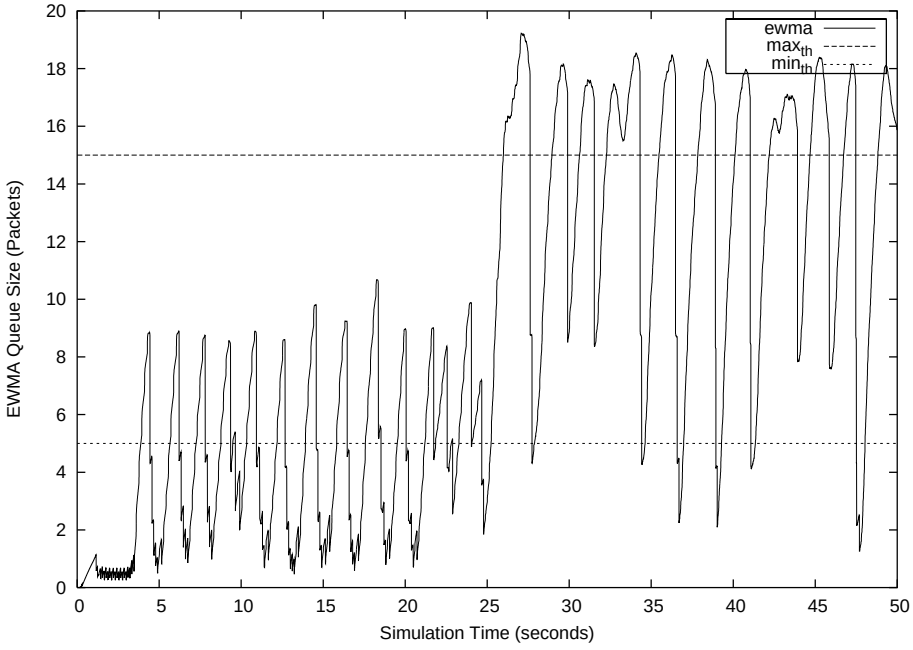

Figure 6.58: EWMA queue size of Hybrid RED with $\{\theta, \xi\} = \{4, 2.0\}$ in simulation topology shown in Figure 6.2.

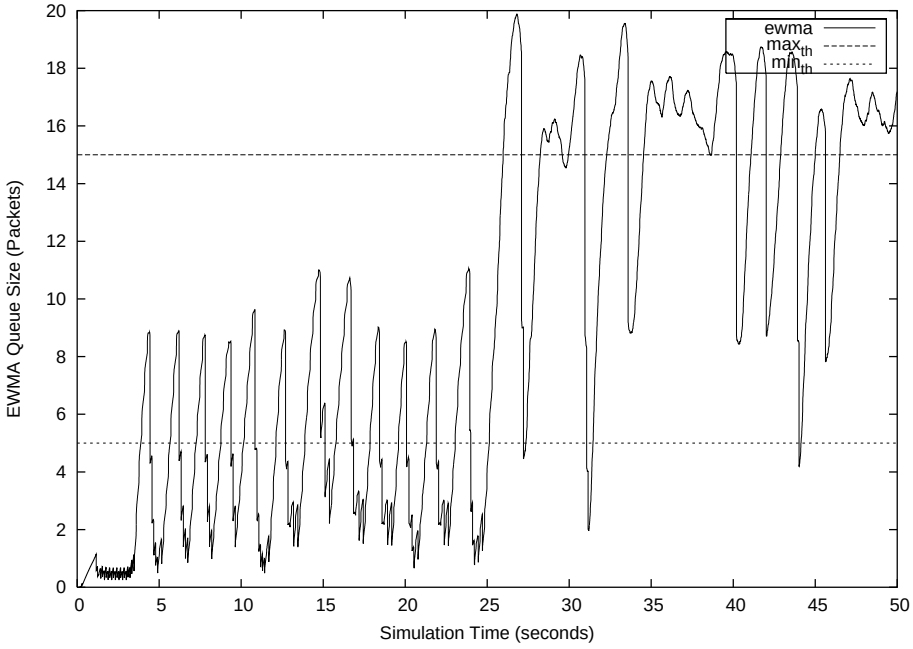

Figure 6.59: EWMA queue size of Hybrid RED with $\{\theta, \xi\} = \{5, 2.0\}$ in simulation topology shown in Figure 6.2.

Bibliography

Braden, B., Clark, D., et al., April 1998. Recommendations on Queue Management and Congestion Avoidance in the Internet. Network Working Group, Request for Comments: 2309 Category: Informational.

DARPA/VINT, LBNL, Xerox, UCB, USC/ISI, 2004. Network simulator ns 2.27. Information available at http://www.isi.edu/nsnam/ns/, code available at http://www.isi.edu/nsnam/dist/.

Feng, W.-C., Kandlur, D. D., Saha, D., Shin, K. G., March 1999. A Self Configuring RED Gateway. Proceedings of the IEEE INFOCOM 1999 , 1320 – 1328Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies.

Floyd, S., November 1997. RED: Discussions of setting parameters. Email available at http://www.aciri.org/floyd/REDparameters.txt.

Floyd, S., March 2000. Recommendation on using the gentle_ variant of RED. Technical note available at http://www.aciri.org/floyd/red/gentle.html.

Floyd, S., Fall, K., February 1997. Router Mechanisms to Support End-to-End Congestion Control. Tech. rep., Lawrence Berkeley National Laboratory, Berkeley, CA, USA, network Research Group.

Floyd, S., Fall, K., August 1999. Promoting the Use of End-to-End Congestion Control in the Internet. IEEE/ACM Transactions on Networking 7 (4), 458–472.

Floyd, S., Gummadi, R., Shenker, S., August 2001. Adaptive RED: An Algorithm for Increasing the Robustness of RED's Active Queue Management. Unpublished draft available at http://www.icir.org/floyd/papers/adaptiveRed.pdf.

Floyd, S., Jacobson, V., August 1993. Random Early Detection Gateways for Congestion Avoidance. IEEE/ACM Transactions on Networking 1 (4), 397–413.

Floyd, S., Jacobson, V., April 1994. The Synchronization of Periodic Routing Messages. IEEE/ACM Transactions on Networking 2 (2), 122–136.

Haider, A., Sirisena, H., Pawlikowski, K., Ferguson, M. J., November-December 2001. Congestion Control Algorithms in High Speed Telecommunication Networks. Proceedings of 36th Annual ORSNZ , 88–97Christchurch, New Zealand. http://www.mang.canterbury.ac.nz/orsnz/conf2001/papers/Haider.pdf.

Hostetter, G. H., 1988. Digital Control System Design. Holt, Rinehart and Winston, Inc., 111 Fifth Avenue, New York 10003, USA.

Jacobson, V., Nichols, K., Poduri, K., September 1999. RED in a Different Light. Unpublished draft available at http://www.cnaf.infn.it/∼ferrari/papers/ispn/red_light_9_30.pdf.

Lin, D., Morris, R., October 1997. Dynamics of Random Early Detection. ACM Computer Communication Review 27 (4), 127–136, cannes, France.

May, M., Diot, C., Lyles, B., Bolot, J., August 2000. Influence of Active Queue Management Parameters on Aggregate Traffic Performance. Tech. Rep. RR-3995, INRIA, France, ftp://ftp.inria.fr/INRIA/publication/publi-pdf/RR/RR-3995.pdf.

Misra, A., Ott, T., Baras, J., June 2001. Effect of Exponential Averaging on the Variability of a RED Queue . Proceedings of IEEE International Conference on Communications, ICC 2001. 6, 1817–1823.

Ott, T., December 1999. On the Ornstein-Uhlenbeck Process with Delayed Feedback. Unpublished draft available at http://web.njit.edu/∼ott/Papers/.

Ramakrishnan, K., Floyd, S., Black, D., September 2001. The Addition of Explicit Congestion Notification (ECN) to IP. Network Working Group, RFC 3168 Category: Standards Track.

Rizzo, L., June 1997. RED and non-responsive flows. end2end-interest mailing list, ftp://ftp.isi.edu/end2end/end2end-interest-1997.mail;type=a.

Sirisena, H., Haider, A., Pawlikowski, K., November 2002. Auto-Tuning RED for Accurate Queue Control. Proceedings of the IEEE GLOBECOM'02 2, 2010–2015, taipei, Taiwan. http://www.globecom2002.com.tw/.

Stoica, I., Shenker, S., Zhang, H., September 1998. Core Stateless Fair Queueing: Achieving Approximatley Fair Bandwidth Allocations in High Speed Networks. Proceedings of the ACM SIGCOMM'98 28 (4), 118–130.

Yang, C.-Q., Reddy, A. V. S., July-August 1995. A Taxonomy for Congestion Control Algorithms in Packet Switching Networks. IEEE Network 9 (4), 34–45.

Young, P., 1984. Recursive Estimation and Time-Series Analysis. Sringer-Verlag, Springer-Verlag Berlin, Heidelberg, Germany, iSBN 3-540-13677-0.

Zheng, B., Atiquzzaman, M., February 2001. Low Pass Filter/Over Drop Avoidance (LPF/ODA): An algorithm to improve the performance of RED gateways . Tech. Rep. CS-TR-01-001, University of Oklahoma, USA, http://www.icir.org/floyd/red.html.

Zheng, B., Atiquzzaman, M., December 2002. Low pass filter/over drop avoidance (LPF/ODA): an algorithm to improve the response time of RED gateways. Wiley International Journal of Communication Systems 15 (10), 899–906, http://www3.interscience.wiley.com/cgi-bin/jissue/102522518.

Zhu, C., Yang, O. W. W., Aweya, J., Ouellette, M., Montuno, D. Y., June 2002. A Comparison of Active Queue Management Algorithms Using the OPNET Modeler. IEEE Communications Magazine 40 (6), 158–167.