The General Renovation as the Active Queue Management Mechanism. Some Aspects and Results

Viana C. C. Hilquias[1(✉)], I. S. Zaryadov[1,2], V. V. Tsurlukov[1], T. A. Milovanova[1], E. V. Bogdanova[1], A. V. Korolkova[1], and D. S. Kulyabov[1,3]

[1] Department of Applied Probability and Informatics, Peoples' Friendship University of Russia (RUDN University), Miklukho-Maklaya street 6, Moscow 117198, Russia
{1042195028,zaryadov-is,1032181900,milovanova-ta, korolkova-av,kulyabov-ds}@rudn.ru

[2] Institute of Informatics Problems, FRC CSC RAS, IPI FRC CSC RAS, 44-2 Vavilova Street, Moscow 119333, Russia

[3] Laboratory of Information Technologies, Joint Institute for Nuclear Research, Joliot-Curie 6, Dubna, Moscow region 141980, Russia

Abstract. This work is devoted to some aspects of using the general renovation (the definition and brief overview are given) as the active queue management mechanism (like RED (Random Early Detection) algorithms).

The attention is paid to the queuing system in which a threshold mechanism and the general renovation mechanism are implemented. This allows to adjust the number of packets in the system by dropping (resetting) the packets from the queue depending on the ratio of a certain control parameter with specified thresholds. But in contrast to standard RED-like systems, a possible reset occurs not at the time of arriving of the next packets in the system, but at the time of the end of service on the device (server). Numerical results for the main probability characteristic (stationary loss probability) are presented.

Keywords: Random early detection · Active queue management · Queuing system · General renovation · Threshold mechanism

The publication has been prepared with the support of the "RUDN University Program 5–100" (A. V. Korolkova—the analysis of RED algorithms, Viana C. C. Hilquias—mathematical model development, V. V. Tsurlukov — the scientific review preparation, E. V. Bogdanova—simulation modelling and numerical analysis). Also the publication has been funded by Russian Foundation for Basic Research (RFBR) according to the research project No. 19-01-00645 (D. S. Kulyabov—simulation model of RED algorithms development), No. 18-07-00692 (I. S. Zaryadov—statement of the problem, the mathematical model development, T. A. Milovanova—the mathematical model development) and No. 19-07-00739 (I.S. Zaryadov, T.A. Milovanova—numerical analysis based on the obtained analytical results).

© Springer Nature Switzerland AG 2019
V. M. Vishnevskiy et al. (Eds.): DCCN 2019, CCIS 1141, pp. 488–502, 2019.
https://doi.org/10.1007/978-3-030-36625-4_39

1 Introduction

The problem of mitigation of congestion and congestion avoidance in modern communication networks is the actual task for researches and practitioners, and, as may be seen [1], this problem does not have a satisfying solution.

According to RFC 7567 [1] active queue management (AQM) is considered as a best practice of network congestion avoidance (reducing) in Internet routers. The active queue management is a based on some rules (algorithms such as random early detection (RED) [2–20], Random Exponential Marking (REM), Blue [21] and stochastic fair Blue (SFB), Adaptive virtual queue algorithm [18–20,22], Explicit Congestion Notification (ECN) [3,23], or controlled delay (CoDel) [24–26]) technique of intelligent drop of network packets inside a buffer associated with a network interface controller (NIC), when that buffer becomes full or gets close to becoming full.

But, as was mentioned before, the problem of congestion avoidance is still actual [31–36], so there exists the IETF working group on ‘’Active Queue Management and Packet Scheduling” [27], where some more novel AQM algorithms are investigated and standardised. A numerous number of AQM schemes have been proposed [19,28–37]. The performance analysis of the most of them is performed by simulation (for example, [34,38]) and the bridges between the available use-case results and analytic results, as well as between the available analytic results are very few (see, for example, [39–43]).

We will consider only the case of RED algorithm and some its modifications. RED has the ability to absorb bursts and also is simple, robust and quite effective at reducing persistent queues [2–5].

In this paper the mathematical model of RED-like algorithm with thresholds will be presented. The word “RED-like” is used because in contrast to standard RED algorithm, when a possible reset occurs at the time of arriving of the next packet in the system [2] and the control parameter is an exponentially weighted average queue length as for classic RED [2], our model is based on completely different idea: the decision about a possible packet drop is synchronised with the service completions. Thus we will use the so called renovation mechanism [44,57–60,63,64].

Some experimental results [61,62] in this direction show that the use of the renovation mechanism in the finite-capacity single server queues under heavy overload conditions allows one to achieve at least the same performance level, as the one guaranteed by the classical random early detection scheme.

The structure of the article is following: the Sect. 2 gives the brief description of the classic RED algorithm, the Sect. 3 is devoted to the general renovation mechanism, some results concerning RED and renovation, and our mathematical model based on general renovation mechanism with thresholds. The last section concludes the paper with the short discussion.

2 The Brief Description of RED Algorithm Module

The classic RED [2] is a queueing discipline with two thresholds (Q_{min} and Q_{max}) and a low-pass filter to calculate the average queue size $\hat{Q}$:

$$\hat{Q}_{k+1} = (1 - w_q)\hat{Q}_k + w_q\hat{Q}_k, \quad k = 0, 1, 2, \ldots, \tag{1}$$

where w_q, $0 < w_q < 1$ is a weight coefficient of the exponentially weighted moving-average and determines the time constant of the low-pass filter. In the optimal bounds for w_q are presented.

RED monitors the average queue size and drops (or marks when used in conjunction with ECN) packets based on statistical probabilities $p(\hat{Q})$ [2].

$$p(\hat{Q}) = \begin{cases} 0, \quad 0 \leq \hat{Q} \leq Q_{min}, \\ \dfrac{\hat{Q} - Q_{min}}{Q_{max} - Q_{min}} p_{max}, \quad Q_{min} < \hat{Q} \leq Q_{max}, \\ 1, \quad \hat{Q} > Q_{max}, \end{cases} \tag{2}$$

p_{max} is the maximum level of packages to be dropped (marked or reset).

RED is more fair than tail drop when the incoming packet is dropped only if the buffer is full. Also RED does not possess a bias against bursty traffic that uses only a small portion of the bandwidth. But, as shown in [7], RED has a number of problems, one of which is that it need tuning and has a little guidance on how to set configuration parameters.

The RED modifications are well presented in [37], but we will specify some of them.

Weighted RED (WRED) [8]—in this algorithm different probabilities for different types of traffic with different priorities (IP precedence, DSCP) and/or queues may be defined. The modification of WRED is Distributed Weighted RED (DWRED)[9].

Adaptive RED or active RED (ARED) [10]—was designed in order to make RED algorithm (based on the observation of the average queue length) more or less aggressive. If the average queue length $\hat{Q}$ oscillates around Q_{min} minimum threshold then early detection considers to be aggressive. If the average queue length $\hat{Q}$ oscillates around Q_{max} threshold then early detection is being too conservative. The drop probability is changed by the algorithm according to how aggressively it senses it has been discarding traffic.

Robust RED (RRED) [11]—is proposed for TCP throughput improvement against DoS (Denial-of-Service) attacks, especially LDoS (Low-rate Denial-of-Service) attacks. The basic idea behind the RRED is to detect and filter out LDoS attack packets from incoming flows before they feed to the RED algorithm. When loss of a sent packet is detected by the source then there will be a transmit delay, so a packet which was sent within a short-range after a loss detection will be suspected to be an attacking packet. This is the basic idea of the detection algorithm of Robust RED (RRED).

EASY RED [12]—is a simpler variant of RED. the drop probability is defined not by average queue length but by instantaneous queue length. The reason is to inform the sender about congestion as soon as possible. The EASY RED parameters are the minimum threshold Q_{min} and the drop probability p_{drop}, which is a constant and used only when the instantaneous queue length is greater or equal to Q_{min}.

Stabilized RED (SRED) [13]—aims at stabilizing buffer occupation by estimating the number of active connections in order to set the drop probability as a function of the number of the active flows and of the instantaneous queue length.

Flow RED (FRED) [14]—uses per-active-flow accounting (based on minimum and maximum limits on the packets that a each flow may have in the queue) to impose on each incoming flow a loss rate (depends on the degree of buffer usage by a flow), it also uses a more aggressive drop against the flows that violates the maximum bound. The state information about active connections also needs to be maintained in the routers.

Balanced RED (BRED) [15]—is proposed to regulate the bandwidth of a flow also by doing per-active-flow accounting for the buffer, similar to FRED but with a different approach: in BRED, two variables (the measures of the packet number for one flow in the buffer and the packet number accepted from this flow since the previous packet dropping) for each flow having packets in the buffer are maintained, which are. As a result the decision of packet drop or acceptance is based on before mentioned two flow state variables.

Dynamic RED (DRED) [16]—is proposed to discard packets with a load dependent probability. The drop probability is updated by employing an integral controller (the input of the controller is the difference between the average queue length and the target buffer level, the output is the drop probability).

The more information about RED and its modifications (Gentle RED (GRED), RED with In and Out (RIO), WRED with thresholds (WRT), Exponential RED (EXPRED), Double Slope RED (DSRED), Random Early Dynamic Detection (REDD)), as other AQM algorithms (Random Exponential Marking (REM), Blue [21] and stochastic fair Blue (SFB), Adaptive virtual queue algorithm) is available at Sally Floyd webpage [17], or in [18–20], new approaches to AQM [22].

3 General Renovation as an Active Queue Management Scheme

3.1 The Definition of General Renovation Mechanism

The renovation mechanism was first defined in the paper [44] and the idea of this mechanism was following: at the moment of the end of its service the packet on the server may either just leave the system with some non-zero probability p, or may empty the buffer with the renovation probability $q = 1 - p$. In [44] the steady-state probability distributions for several types of queueing systems were presented.

Some applications of the renovation mechanism in finance other fields were shown in [45]. So, the queues with renovation are similar to queues with disasters (or negative customers), when the incoming flow of signals cause the buffer to drop some or all the packets, or to queues with unreliable servers, which cause the packet dropping ([46–56]).

In [57] the mathematical model of renovation mechanism with repeated service (or feedback) was proposed by P. P. Bocharov. It means that the served packet after emptying the buffer with probability q enters the server for another round of service. The main characteristics in matrix-analytical form were obtained.

Later on the renovation mechanism was further generalised by Pechinkin [58], who proposed the mathematical model of general renovation: at the end of service the packet discards from the buffer of capacity $0 < r < \infty$ exactly i, $i \geq 1$, other packets with probability $q(i)$ and leaves the system, or just leaves the system without any effect on it with the complementary probability $p = 1 - \sum_{i=1}^{r} q(i)$. In [58–60] the multiserver $GI|M|n|\infty$ and $GI|M|n|r$ queueing systems with different renovation and service disciplines were studied. The general renovation with feedback (the retrial queueing system with general renovation and recurrent input flow) was investigated in [63,64].

3.2 The Mathematical Model of RED-like Algorithm by Queueing System with Renovation and Thresholds

In this part of the article we will discuss another model of RED-like algorithm based on queueing systems with renovation and thresholds.

In [65] the queueing system with two thresholds ($q_{\min} < q_{\max}$), recurrent input flow of packets and exponentially identically distributed service times on C servers with renovation mechanism was considered.

The idea of mechanism of renovation with thresholds is following. At the moment of the end of a packet service the current queue length $\tilde{q}$ is compared with thresholds $q_{\min}$ and $q_{\max}$, and if $\tilde{q} \leq q_{\min}$ then no one of the packets from the buffer is dropped. If $q_{\min} + 1 \geq \tilde{q} \leq q_{\max}$ then the last packet in the buffer is dropped with probability $p(\tilde{q})$, $0 < p(\tilde{q}) < p_{\max}$. If $\tilde{q} \geq q_{\max} + 1$ also the last packet in the buffer is dropped but with maximal probability $p_{\max}$.

The steady-state probability distribution of packets in the system(for the imbedded upon arrival moments Markov chain) were obtained in geometric form:

$$p_0 = \sum_{j=1}^{C-1} p_{j-1} A_{j,0} + p_{C-1} A_0^*, \tag{3}$$

$$p_i = \sum_{j=i-1}^{C-1} p_j A_{j+1,i} + p_{C-1} A_i^*, \quad i = \overline{1, C-1}, \tag{4}$$

$$p_{C+j} = p_{C-1} \prod_{m=0}^{j} g_{q_{\min}-m}, \quad j = \overline{0, q_{\min} - 1}, \tag{5}$$

$$p_{C+q_{\min}+j} = p_{C-1} \prod_{m=1}^{q_{\min}} g_m \prod_{l=0}^{j} \tilde{g}_{q_{\max}-q_{\min}-l}, \quad j = \overline{0, q_{\max} - q_{\min} - 1}, \tag{6}$$

$$p_{C+q_{\max}+j} = p_{C-1} \hat{g}^{j+1} \prod_{m=1}^{q_{\min}} g_m \prod_{l=0}^{q_{\max}-q_{\min}} \tilde{g}_l, \quad j \geq 0, \tag{7}$$

where $A_(i,j)$, $i,j \geq 0$ are elements of the transition probability matrix of the embedded Markov chain [65], A_i^*, $i = \overline{0, C-1}$ are auxiliary values [65], $\hat{g}$ is the unique solution of the equation $\hat{g} = \alpha\left(C\mu(1-\hat{g})\right)$ which belongs to the interval $(0;1)$, $\alpha(\cdot)$ is the Laplace-Stieltjes transformation, the values g_i and $\tilde{g}_i$ are defined by

$$g_i = \frac{\alpha(C\mu)}{L_i}, \quad i = \overline{1, q_{\min}}, \quad \tilde{g}_i = \frac{C\mu}{K_i}, \quad i = \overline{1, q_{\max} - q_{\min} - 1}, \tag{8}$$

and L_i, $i = \overline{1, q_{\min}}$, K_i, $i = \overline{1, q_{\max} - q_{\min} - 1}$ are also fully defined in [65].

The probability p^{loss} that the incoming packet will be dropped from the system by one of the served packets is:

$$p^{loss} = p_{C-1}\left(1 - \alpha(C\mu)\right) \left(\sum_{i=q_{\min}}^{q_{\max}-1} p(\hat{q}) \prod_{m=1}^{q_{\min}} g_m \prod_{l=0}^{i-q_{\min}} \tilde{g}_{q_{\max}-q_{\min}-l} + \right.$$
$$\left. + \prod_{m=1}^{q_{\min}} g_m \prod_{l=0}^{q_{\max}-q_{\min}} \tilde{g}_l \frac{\hat{g}}{1-\hat{g}} \right). \tag{9}$$

The probability p^{serv} that the incoming packet will be served is:

$$p^{serv} = 1 - p^{loss}. \tag{10}$$

The mean waiting time of a served packet w^{serv} is

$$w^{serv} = \frac{p_{C-1}}{p^{serv}} \left(\sum_{i=0}^{q_{\min}-1} \frac{i+1}{C\mu} \prod_{m=0}^{i} g_{q_{\min}-m} + + \prod_{m=1}^{q_{\min}} g_m \sum_{i=q_{\min}}^{q_{\max}-1} \prod_{l=0}^{i-q_{\min}} \tilde{g}_{q_{\max}-q_{\min}-l} \right.$$
$$\left(\frac{(i+1)(1-p(\tilde{q}))}{C\mu} - \alpha^{(1)}(C\mu) + \frac{(i+1)\alpha(C\mu)}{C\mu} \right) +$$
$$\left. + \prod_{m=1}^{q_{\min}} g_m \prod_{l=0}^{q_{\max}-q_{\min}} \tilde{g}_l \left(\frac{\alpha(C\mu)\hat{g}q_{\max}}{C\mu(1-\hat{g})} + \frac{\alpha(C\mu)\hat{g}}{C\mu(1-\hat{g})^2} - \frac{\alpha^{(1)}(C\mu)\hat{g}}{(1-\hat{g})} \right) \right). \tag{11}$$

The recommendations on the optimal values of $(q_{\min} < q_{\max})$ (based on the numerical analysis of the obtained characteristics) were similar as in [5].

3.3 The Mathematical Model of RED-like Algorithm by Queueing System with General Renovation and One or Two Thresholds

But we want to consider the model with general renovation — when a group of packets may be dropped from the buffer. For example, the authors work on the $G|M|1|\infty$ system with only one threshold $q_{\min}$. If the current queue size $\tilde{q} \leq q_{\min}$, then served packet just leave the system. But if $q_{\min} + 1 \geq \tilde{q}$ then three different types of dropping mechanism may be applied:

1. with probability $q(i)$ $(i \geq 1)$ the group of i packets from the buffer is dropped (if there are less than i packets in buffer, the buffer will be emptied);
2. with probability $q(i)$ $(i \geq 1)$ the group of i packets from the buffer is dropped if there were $q_{\min} + i$ packets or only $q_{\min}$ packets will remain in the buffer;
3. the virtual threshold q^* is introduced and with probability $q(i)$ $(i \geq 1)$ the group of i packets from the buffer is dropped if $\tilde{q} - i \leq q^*$ or only q^* packets will remain in the buffer.

The minus of the first drop mechanism is that too many packets may be dropped. The minus of the second drop mechanism is that the buffer may remain overflowed. The third mechanism is the combination of the previous ones without minuses of the previous ones, but it may be difficult for analytical investigation.

Our goal is to construct mathematical models for all three cases and to compare such characteristics as the probability p^{loss} of an arbitrary arrived packet being dropped from the system, and mean sojourn times w^{loss} and w^{serv} for lost (dropped) packets and served packets.

For the first model we obtained the steady-state probability distribution of packets p_i $(i \geq 0)$ in the system(for imbedded Markov chain), some probabilities are represented by geometric form (when the threshold q is overcomed):

$$p_i = \sum_{j=i-1}^{q} p_j (-\mu)^{j+1-i} \alpha^{(j+1-i)}(\mu) + \\ + p_{q+1} g^{i-q-2} \left(g - \alpha(\mu) - \int_0^{\infty} A(g,x) e^{-\mu x} dA(x) \right), \\ i = \overline{1, q+1}, \quad (12)$$

$$p_i = p_{q+1} g^{i-(q+1)}, i > q+1, \quad (13)$$

$$p_0 = \sum_{i=0}^{\infty} p_i p_{i,0}, \quad (14)$$

$$p_{i,0} = 1 - \sum_{j=1}^{i} (-\mu)^j \alpha^{(j)}(\mu), 0 < i \leq q, p_{i,0} = 1 - \sum_{j=1}^{i} \pi(j, i-j) \alpha^{(j)}(\mu), i > q. \quad (15)$$

where g is the unique solution of the equation

$$g = \alpha\left(\mu(1 - gQ(g))\right), \tag{16}$$

and belongs to interval $(0;1)$, $\alpha(s)$ is the Laplase-Stieltjes transformation of interrarrival time distribution function $A(x)$, $Q(g)$ is the probability generating function

$$Q(l,g) = \sum_{k=0}^{\infty} \pi(l,k) g^k = Q^l(g), \quad Q(g) = p + \sum_{k=1}^{\infty} \pi(1,k) g^k. \tag{17}$$

for probabilities $\pi(l,k)$, $l \geq 1, k \geq 0$, that l packets will be served and k packets will be dropped from the buffer.

The probabilities $\pi(l,k)$, $l \geq 1$, $k \geq 0$, may be defined by following. If k packets are served and none of the packets are dropped from the buffer, then

$$\pi(l,0) = p^l, \quad l \geq 1, \quad \pi(0,k) \equiv 0, \quad k \geq 1, \tag{18}$$

because the packets may be dropped from the queue only at the moment of the end of the service, but

$$\pi(0,0) \equiv 1, \tag{19}$$

if the service on the server has not ended, then no packet can be dropped from the queue.

$$\pi(1,k) = q(k), \quad k \geq 1, \tag{20}$$

is the probability that a packet at the moment of the end of the service will drop from the queue k other packets. And the general formula is

$$\pi(l,k) = \sum_{n=0}^{k} \pi(1,n)\pi(l-1,k-n), \quad l \geq 1, \quad k \geq 0. \tag{21}$$

$$A(g,x) = \sum_{l=1}^{q+1-i} \frac{(\mu x g)^l}{l!} \sum_{j=0}^{q+1-i-l} \pi(l,j) g^j. \tag{22}$$

Also the probability that the arriving packet will be dropped and sojourn time characteristics for dropped packets are obtained in form of integral equations.

$$p^{loss} = \sum_{i=1}^{\infty} p_{i,0}^{loss} p_i, \tag{23}$$

where $p_{i,j}^{loss}$ is the the probability that the “selected” packet will be dropped if there are i, $i \geq 1$, packets before it and j, $j \geq 0$ packets behind it; $p_{i,0}^{loss}$ is the probability that the arriving packet will be dropped if there are i, $i \geq 1$, packets in the system.

$$p_{i,j}^{loss} = \int_0^{\infty} \int_0^{x} \sum_{k=0}^{i} \frac{(\mu y)^k}{k!} e^{-\mu y} dA(y) p_{i-k,j+1}^{loss}(x-y) dx, \quad 0 \leq i+j \leq q, \tag{24}$$

$$p_{0,j}^{loss} \equiv 0, j \geq 0, \tag{25}$$

because the packet could not be dropped being on service.

$$p_{i,j}^{loss} = \int_0^\infty \bar{A}(x) \sum_{m=1}^{i} \frac{\mu^m x^{m-1}}{(m-1)!} e^{-\mu x} \pi^*(m, i+j) dx + \\ + \int_0^\infty \int_0^x \sum_{m=0}^{i} p^m \frac{(\mu y)^m}{m!} e^{-\mu y} dA(y) p_{i-m,j+1}^{loss}(x-y) dx, \quad i+j > q, \tag{26}$$

where $\pi^*(m, i)$ is the auxiliary probability that m, $m \geq 1$, served packets will completely empty the buffer of the system if there were i, $i \geq 0$, packets in it:

$$\pi^*(1, i) = \sum_{k=i}^{\infty} q(k), \quad \pi^*(m, i) = \sum_{k=0}^{i} \pi(1, k) \pi^*(m-1, i-k), \quad m > 1, i \geq 0. \tag{27}$$

The mean value of packets in the system is defined by the following formula:

$$N = \sum_{i=0}^{\infty} i p_i = \sum_{i=1}^{q} i p_i + p_{q+1} \frac{q(1-g)+1}{(1-g)^2}. \tag{28}$$

The probability distribution of the time in the system for dropped packet $W^{loss}(x)$ may de derived by the same way as p^{loss}.

$$W^{loss}(x) = \sum_{i=0}^{\infty} W_{i,0}^{loss}(x) p_i, \tag{29}$$

where $W_{i,0}^{loss}(x)$ is the probability that the packet will be dropped from the buffer for time less than x if there were i, $i \geq 0$, other packets in the system at the arrival moment. In terms of Laplace-Stieltjes (29) takes form:

$$\omega^{loss}(s) = \sum_{i=0}^{\infty} \omega_{i,0}^{loss}(s) p_i, \tag{30}$$

where

$$\omega_{i,j}^{loss}(s) = \sum_{m=1}^{i} \frac{(-\mu)^m \alpha^{(m)}(\mu s)}{m!} \omega_{i-m,j+1}^{loss}(s), \quad i+j \leq q, \tag{31}$$

$$\omega_{i,j}^{loss}(s) = \sum_{m=1}^{i} \frac{(-1)^{m-1} \mu^m \bar{\alpha}^{(m-1)}(\mu s)}{(m-1)!} \pi^*(m, i+j) + \\ + \sum_{m=0}^{i} \frac{(-\mu p)^m \alpha^{(m)}(\mu s)}{m!} \omega_{i-m,j+1}^{loss}(s), \quad i+j > q. \tag{32}$$

3.4 The Comparison of RED Algorithm with General Renovation and General Renovation with Feedback

In this section we will compare the loss probability p^{loss} (the probability of packet being dropped from the system) for RED and TailDrop algorithms (the values of p^{loss} are presented in [40,43]) and values of the probability p^{loss}, obtained by formulas derived for queueing system with general renovation [61,62], queueing system with general renovation and feedback [63,64], renovation with two thresholds (Sect. 3.2) and general renovation with one threshold (Sect. 3.3). Also we will use results obtined by the members of our authors group in their Master's thesises.

Table 1. The values of p^{loss} for Taildrop, RED, general renovation and general renovation with feedback

Loss probability					
Taildrop	RED	renov	ren-fd	ren-2-th	ren-1-th.
$\rho = 0.5$					
0	0.002	0.002	0.0002	0.00021	0.000205
$\rho = 1$					
0.051	0.091	0.104	0.11	0.067	0.074
$\rho = 2$					
0.500	0.500	0.502	0.54	0.503	0.500
$\rho = 3$					
0.667	0.667	0.667	0.71	0.668	0.666

As can be see from the Table 1, according to the values of the p^{loss}, all types of renovation mechanism can perform as good as RED in the wide range of the offered load ρ, but for the case of general renovation fine and accurate tuning of general renovation probabilities $q(i)$ is required,

4 Conclusion

Even though the idea behind the renovation-type AQM is completely different from the idea behind RED-type AQM, renovation-type AQM may allow one to achieve in some cases at least the same system performance level as guaranteed by RED-type AQM.

The presented numerical experiments show that the results remain qualitatively the same for RED-type AQM with other dropping functions. Being defined by N parameters, the renovation mechanism is very flexible and this constitutes its strength and weakness. By varying the values of the renovation probabilities $q(i)$, it is possible to carry out conditional optimisation, but good searching procedures are required here.

Implementation of the renovation as a packet dropping mechanism requires a priori tuning and/or operational configuration of its parameters. Thus, whether it is appropriate to use renovation as a packet dropping mechanism or not in practice heavily depends on the use case. Although the tuning of the renovation parameters qi can be made on the fly during operation, with respect to the recommendations of the RFC 7567 [1], renovation mechanism is not the proper choice for the network congestion control unless simple recommendations on how to set up the renovation parameters are given. We believe this can be done based on more deep and insightful numerical experiments.

At the end, it is worthy of mention that there is another approach to the analysis of behaviour of networks with burst traffic — the method based on hysteretic thresholds load control [66–69] and it will interesting to investigate systems with hysteretic control of renovation probabilities, especially for the case of Markov arrival process [69]. The authors will try to combine the method of hysteretic thresholds load control with renovation mechanism.

References

1. Baker, F., Fairhurst, G.: IETF recommendations regarding active queue management. RFC 7567. Internet Engineering Task Force. https://tools.ietf.org/html/rfc7567. Accessed 29 Apr 2019
2. Floyd, S., Jacobson, V.: Random early detection gateways for congestion avoidance. IEEE/ACM Trans. Netw. **4**(1), 397–413 (1993). https://doi.org/10.1109/90.251892
3. Ramakrishnan, K., Floyd, S., Black, D.: The addition of explicit congestion notification (ECN) to IP. RFC 3168. Internet Engineering Task Force. https://tools.ietf.org/html/rfc3168. Accessed 29 May 2019
4. Floyd, S., Gummadi, R., Shenker, S.: Adaptive RED: An Algorithm for Increasing the Robustness of RED's Active Queue Management (2001). http://www.icir.org/floyd/papers/adaptiveRed.pdf
5. Floyd, S.: RED: Discussions of Setting Parameters (1997). http://www.aciri.org/floyd/REDparameters.txt
6. Korolkova, A.V., Zaryadov, I.S.: The mathematical model of the traffic transfer process with a rate adjustable by RED. In: International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT), pp. 1046–1050. IEEE. Moscow. Russia (2010) https://doi.org/10.1109/ICUMT.2010.5676505
7. Jacobson, V., Nichols, K., Poduri, K.: RED in a Different Light (2019). http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.22.9406. Accessed 1 Sept 2019
8. Class-Based Weighted Fair Queueing and Weighted Random Early Detection (2019). http://www.cisco.com/c/en/us/td/docs/ios/12_0s/feature/guide/fswfq26.html. Accessed 1 Sept 2019
9. Cisco IOS Quality of Service Solutions Configuration Guide, Release 12.2. https://www.cisco.com/c/en/us/td/docs/ios/qos/configuration/guide/12_2sr/qos_12_2sr_book.pdf. Accessed 1 Sept 2019
10. Floyd, S., Gummadi, R., Shenker, S.: Adaptive RED: An Algorithm for Increasing the Robustness of RED's Active Queue Management (2001). http://www.icir.org/floyd/papers/adaptiveRed.pdf. Accessed 1 Sept 2019

11. Changwang, Z., Jianping, Y., Zhiping, C., Weifeng, C.: RRED: robust RED algorithm to counter low-rate Denial-of-Service attacks. IEEE Commun. Lett. **14**(5), 489–491 (2010). https://doi.org/10.1109/LCOMM.2010.05.091407
12. Grieco, L.A., Mascolo, S.: TCP westwood and easy RED to improve fairness in high-speed networks. In: Carle, G., Zitterbart, M. (eds.) PfHSN 2002. LNCS, vol. 2334, pp. 130–146. Springer, Heidelberg (2002). https://doi.org/10.1007/3-540-47828-0_9
13. Ott, T.J., Lakshman, T.V., Wong, L.H.: SRED: stabilized RED. In: Proceedings IEEE INFOCOM 1999, vol. 3, pp. 1346–1355. IEEE, New York, NY, USA (1999). https://doi.org/10.1109/INFCOM.1999.752153
14. Lin, D., Morris, R.: Dynamics of random early detection. Comput. Commun. Rev. **27**(4), 127–137 (1997)
15. Anjum, F.M., Tassiulas, L.: Balanced RED: An Algorithm to Achieve Fairness in the Internet. Technical Research Report (1999). http://www.dtic.mil/dtic/tr/fulltext/u2/a439654.pdf
16. Aweya, J., Ouellette, M., Montuno, D.Y.: A control theoretic approach to active queue management. Comput. Netw. **36**, 203–235 (2001)
17. Sally Floyd Website. http://www.icir.org/floyd/. Accessed 1 Sept 2019
18. Chrysostomoua, C., Pitsillidesa, A., Rossidesa, L., Polycarpoub, M., Sekercioglu, A.: Congestion control in differentiated services networks using fuzzy-RED. Control Eng. Pract. **11**, 1153–1170 (2003)
19. Feng W.-C.: Improving Internet Congestion Control and Queue Management Algorithms. http://thefengs.com/wuchang/umich_diss.html
20. Al-Raddady, F., Woodward, M.: A new adaptive congestion control mechanism for the internet based on RED. In: 21st International Conference on Advanced Information Networking and Applications. AINAW 2007 Workshops (2007)
21. Feng, W., Kandlur, D.D., Saha, D., Shin, K.G.: BLUE: A New Class of Active Queue Management Algorithms. UM CSE-TR-387-99 (1999). https://www.cse.umich.edu/techreports/cse/99/CSE-TR-387-99.pdf
22. Baldi, S., Kosmatopoulos, E.B., Pitsillides, A., Lestas, M., Ioannou, P.A., Wan, Y.: Adaptive optimization for active queue management supporting TCP flows. In: 2016 American Control Conference (ACC), pp. 751–756 (2016)
23. Floyd, S.: TCP and Explicit Congestion Notification. https://www.icir.org/floyd/papers/tcp_ecn.4.pdf. Accessed 29 May 2019
24. Nichols, K., Jacobson, V., McGregor, A., Iyengar, J.: Controlled Delay Active Queue Management. RFC 8289. Internet Engineering Task Force. https://tools.ietf.org/html/rfc8289. Accessed 29 Aug 2019
25. Nichols, K., Jacobson, V.: Controlling Queue Delay ACM Queue. ACM Publishing **10**(5), 2020–2034 (2012). queue.acm.org/detail.cfm?id=2209336
26. Hoeiland-Joergensen, T., McKenney, P., Taht, D., Gettys, J., Dumazet, E.: The flow queue codel packet scheduler and active queue management algorithm. Internet Engineering Task Force (2018). https://tools.ietf.org/html/rfc8290
27. IETF Working Group on Active Queue Management and Packet Scheduling (AQM): Description of the Working Group. http://tools.ietf.org/wg/aqm/charters. Accessed 29 Aug 2019
28. McKenney, P.E.: Stochastic fairness queueing. In: Proceedings of IEEE International Conference on Computer Communications, vol. 2, pp. 733–740. IEEE, San Francisco, CA, USA (1990). https://doi.org/10.1109/INFCOM.1990.91316
29. Adams, R.: Active queue management: a survey. IEEE Commun. Surv. Tutorials **15**(3), 1425–1476 (2013)

30. Paul, A.K., Kawakami, H., Tachibana, A., Hasegawa, T.: An AQM based congestion control for eNB RLC in 4G/LTE network. In: 2016 IEEE Canadian Conference on Electrical and Computer Engineering (CCECE), pp. 1–5. IEEE. Vancouver, BC, Canada (2016). https://doi.org/10.1109/CCECE.2016.7726792
31. Dai, Yu., Wijeratne, V., Chen, Y., Schormans, J.: Channel quality aware active queue management in cellular networks. In: Computer Science and Electronic Engineering (CEEC) 2017, pp. 183–188. IEEE. Colchester, UK (2017). https://doi.org/10.1109/CEEC.2017.8101622
32. Beshay, J.D., Nasrabadi, A.T., Prakash, R., Francini, A.: On active queue management in cellular networks. In: 2017 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS), pp. 384–389. IEEE, Atlanta, GA, USA (2017) https://doi.org/10.1109/INFCOMW.2017.8116407
33. Ali Rezaee, A., Pasandideh, F.: A fuzzy congestion control protocol based on active queue management in wireless sensor networks with medical applications. Wirel. Personal Commun. **98**(1), 815–842 (2018). https://doi.org/10.1007/s11277-017-4896-6
34. Menth, M., Veith, S.: Active queue management based on congestion policing (CP-AQM). In: German, R., Hielscher, K.-S., Krieger, U.R. (eds.) MMB 2018. LNCS, vol. 10740, pp. 173–187. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-74947-1_12
35. Adesh, N.D., Renuka, A.: Adaptive receiver-window adjustment for delay reduction in LTE networks. J. Comput. Netw. Commun. **2019**, 17 (2019). https://doi.org/10.1155/2019/3645717. Article ID 3645717
36. Irazabal, M., Lopez-Aguilera, E., Demirkol, I.: Active queue management as quality of service enabler for 5G networks. In: European Conference on Networks and Communications (EuCNC), pp. 421–426. IEEE, Valencia, Spain (2019). https://doi.org/10.1109/EuCNC.2019.8802027
37. Korolkova, A.V., Kulyabov, D.S., Chernoivanov, A.I.: On the classification of RED algorithms. Bulletin of Peoples' Friendship University of Russia. Series "Mathematics. Information Sciences. Physics". No. 3, 34–46 (2009)
38. Korolkova, A.V., Velieva, T.R., Abaev, P.O., Sevastianov, L.A., Kulyabov, D.S.: Hybrid simulation of active traffic management. In: Proceedings of 30th European Conference on Modelling and Simulation (ECMS), pp. 692–697. ECMS. Regensburg, Germany (2016). https://doi.org/10.7148/2016-0685
39. Bonald, T., May, M., Bolot, J.: Analytic evaluation of RED performance. In: Proceedings IEEE INFOCOM 2000 Conference on Computer Communications, vol. 3, pp. 1415–1424. IEEE, Tel Aviv, Israel (2000). https://doi.org/10.1109/INFCOM.2000.832539
40. Chydzinski, A., Chrost, L.: Analysis of AQM queues with queue size based packet dropping. Int. J. Appl. Math. Comput. Sci. **21**(3), 567–577 (2011). https://doi.org/10.2478/v10006-011-0045-7
41. Zhernovyi, Y., Kopytko, B., Zhernovyi, K.: On characteristics of the $M/G/1/m$ and $M/G/1$ queues with queue-size based packet dropping. J. Appl. Math. Comput. Mech. **13**(4), 163–175 (2014)
42. Chydzinski, A., Mrozowski, P.: Queues with dropping functions and general arrival processes. PLoS One, 11(3) (2016). https://doi.org/10.1371/journal.pone.0150702
43. Konovalov, M.G., Razumchik, R.V.: Numerical analysis of improved access restriction algorithms in a $GI|G|1|N$ system. J. Commun. Technol. Electron. **63**(6), 616–625 (2018). https://doi.org/10.1134/S1064226918060141
44. Kreinin, A.: Queueing systems with renovation. J. Appl. Math. Stochast. Anal. **10**(4), 431–443 (1997). https://doi.org/10.1155/S1048953397000464

45. Kreinin, A.: Inhomogeneous random walks: applications in queueing and finance. In: CanQueue 2003, Fields Institute, Toronto (2003)
46. Gelenbe, E.: Product-form queueing networks with negative and positive customers. J. Appl. Prob. **28**(3), 656–663 (1991). https://doi.org/10.2307/3214499
47. Pechinkin, A.V., Razumchik, R.V.: The stationary distribution of the waiting time in a queueing system with negative customers and a bunker for superseded customers in the case of the LAST-LIFO-LIFO discipline. J. Commun. Technol. Electron. **57**(12), 1331–1339 (2012). https://doi.org/10.1134/S1064226912120054
48. Razumchik, R.V.: Analysis of finite capacity queue with negative customers and bunker for ousted customers using chebyshev and gegenbauer polynomials. Asia-Pacific J. Oper. Res. **31**(04), 1450029 (2014). https://doi.org/10.1142/S0217595914500298
49. Semenova, O.V.: Multithreshold control of the $BMAP/G/1$ queuing system with MAP flow of Markovian disasters. Autom. Remote Control **68**(1), 95–108 (2007). https://doi.org/10.1134/S0005117907010092
50. Li, J., Zhang, L.: $M^X|M|c$ queue with catastrophes and state-dependent control at idle time. Front. Math. China **12**(6), 1427–1439 (2017). https://doi.org/10.1007/s11464-017-0674-8
51. Kim, C., Klimenok, V.I., Dudin, A.N.: Analysis of unreliable $BMAP|PH|N$ type queue with Markovian flow of breakdowns. Appl. Math. Comput. **314**, 154–172 (2017). https://doi.org/10.1016/j.amc.2017.06.035
52. Gudkova, I., et al.: Modeling and analyzing licensed shared access operation for 5G network as an inhomogeneous queue with catastrophes. In: International Congress on Ultra Modern Telecommunications and Control Systems and Workshops, pp. 282–287. IEEE, Lisbon, Portugal (2016). https://doi.org/10.1109/ICUMT.2016.7765372
53. Dudin, A., Klimenok, V., Vishnevsky, V.: Analysis of unreliable single server queueing system with hot back-up server. In: Plakhov, A., Tchemisova, T., Freitas, A. (eds.) EmC-ONS 2014. CCIS, vol. 499, pp. 149–161. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-20352-2_10
54. Krishnamoorthy, A., Pramod, P.K., Chakravarthy, S.R.: Queues with Interruptions: a survey. TOP **22**(1), 290–320 (2014). https://doi.org/10.1007/s11750-012-0256-6
55. Vishnevsky, V.M., Kozyrev, D.V., Semenova, O.V.: Redundant queuing system with unreliable servers. In: International Congress on Ultra Modern Telecommunications and Control Systems and Workshops, pp. 283–286. IEEE, St. Petersburg, Russia (2014) https://doi.org/10.1109/ICUMT.2014.7002116
56. Rykov, V.V., Kozyrev, D.V.: Analysis of renewable reliability systems by markovization method. In: Rykov, V.V., Singpurwalla, N.D., Zubkov, A.M. (eds.) ACMPT 2017. LNCS, vol. 10684, pp. 210–220. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-71504-9_19
57. Bocharov, P.P., Zaryadov, I.S.: Probability distribution in queueing systems with renovation. Bulletin of Peoples' Friendship University of Russia Series "Mathematics Information Sciences Physics". No. 1–2, 15–25 (In Russian) (2007)
58. Zaryadov, I.S., Pechinkin, A.V.: Stationary time characteristics of the $GI/M/n/\infty$ system with some variants of the generalized renovation discipline. Autom. Remote Control **70**(12), 2085–2097 (2009). https://doi.org/10.1134/S0005117909120157
59. Zaryadov, I.S.: Queueing systems with general renovation. In: ICUMT 2009 - International Conference on Ultra Modern Telecommunications, pp. 1–6. IEEE, St. Petersburg, Russia (2009) https://doi.org/10.1109/ICUMT.2009.5345382

60. Zaryadov, I., Razumchik, R., Milovanova, T.: Stationary waiting time distribution in G—M—n—r with random renovation policy. In: Vishnevskiy, V.M., Samouylov, K.E., Kozyrev, D.V. (eds.) DCCN 2016. CCIS, vol. 678, pp. 349–360. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-51917-3_31
61. Konovalov, M., Razumchik, R.: Queueing systems with renovation vs. queues with RED. Supplementary Material (2017). https://arxiv.org/abs/1709.01477
62. Konovalov, M., Razumchik, R.: Comparison of two active queue management schemes through the M/D/1/N queue. Informatika i ee Primeneniya (Informatics and Applications), 12(4), 9–15 (2018). https://doi.org/10.14357/19922264180402
63. Bogdanova, E.V., Zaryadov, I.S., Milovanova, T.A., Korolkova, A.V., Kulyabov, D.S.: Characteristics of lost and served packets for retrial queueing system with general renovation and recurrent input flow. In: Vishnevskiy, V.M., Kozyrev, D.V. (eds.) DCCN 2018. CCIS, vol. 919, pp. 327–340. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-99447-5_28
64. Zaryadov, I., Bogdanova, E., Milovanova, T., Matushenko, S., Pyatkina, D.: Stationary characteristics of the $GI|M|1$ queue with general renovation and feedback. In: 10th International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT), article No. 8631244. IEEE, Moscow, Russia (2019) https://doi.org/10.1109/ICUMT.2018.8631244
65. Zaryadov, I.S., Korolkova, A.V.: The application of model with general renovation to the analysis of characteristics of active queue management with random early detection (RED). T-Comm. No. **7**, 84–88 (2011). (In Russian)
66. Abaev, P., Gaidamaka, Y., Samouylov, K., Pechinkin, A., Razumchik, R., Shorgin, S.: Hysteretic control technique for overload problem solution in network of SIP servers. Comput. Inform. **33**(1), 218–236 (2014)
67. Abaev, P., Khachko, A., Beschastny, V.: Queuing model for SIP server hysteretic overload control with K-state MMPP bursty traffic. In: International Congress on Ultra Modern Telecommunications and Control Systems and Workshops, pp. 495–500. IEEE, St. Petersburg, Russia (2015). https://doi.org/10.1109/ICUMT.2014.7002151
68. Pechinkin, A.V., Razumchik, R.R., Zaryadov, I.S.: First passage times in $M_2^{[X]}|G|1|R$ queue with hysteretic overload control policy. In: AIP Conference Proceedings, vol. 1738, article No. 220007. American Institute of Physics Inc., Rhodes (2016). https://doi.org/10.1063/1.4952006
69. Razumchik, R.: Analysis of finite $MAP|PH|1$ queue with hysteretic control of arrivals. In: International Congress on Ultra Modern Telecommunications and Control Systems and Workshops, article No. 7765373, pp. 288–293. IEEE, Lisbon. Portugal (2016) https://doi.org/10.1109/ICUMT.2016.7765373