INTERNATIONAL JOURNAL OF RESEARCH IN COMPUTER APPLICATIONS AND ROBOTICS
ISSN 2320-7345

ANALYTICAL MODELS TO IMPROVE RED`S PERFORMANCE

Shahram Jamali[1], Samira Hamed[1] and Morteza Analoui[2]
[1]Department of Computer Engineering
University of Mohaghegh Ardabili, Ardebil, Iran
jamali@iust.ac.ir
[2] Department of Computer Engineering
Iran University of Science and Technology, Tehran, Iran

Abstract

One of the most challenging issues for Random Early Detection (RED) algorithm is how to set its parameters to achieve high performance for the dynamic conditions of the network. While original RED uses fixed values for its parameters, the use of Markov model, as a forecasting model to predict the queue length has been a matter of interest. Different approaches have been introduced in this scope. In this paper, we have studied and evaluated different papers using Markov model based on performance criteria such as stability and utilization. Some conclusions are developed regarding the performance of these methods.

.

Keywords: Congestion control; Active queue management; RED; Markov model.

1. Introduction

With growing use of computer networks, importance of congestion control is getting more obvious in maintaining stability and performance. The Internet congestion control includes two different parts: the source algorithm and Active Queue Management (AQM) algorithm. Source algorithm, implemented inside TCP protocol, runs at the end system and controls the sending rate. AQM algorithm runs at the router and sends the congestion feedbacks to the sources.

The most common AQM algorithm is Random Early Detection (RED), which has been widely used in Internet over the past few years. This algorithm mainly addresses the global synchronization problem. This problem happens when all connections “hold back” simultaneously, and then step forward simultaneously. RED is simple, robust and quite effective at reducing persistent queues. However, while it has been used widely and successfully on Internet routers, (Floyd and Jacobson, 1993) offers little guidance on how to set configuration parameters and RED has gained the reputation of being very difficult to tune (Sarker et. al. 2012).RED should stabilize the queue length at a given target to maximize link utilization (Aweya et. al. 2001; Low et. al. 2003). This is usually not obtained according to observations (Tan et al. 2006).

This paper focuses on performance optimization by trying to control oscillations of the average queue length. It considers a network running RED algorithm and designs a Grey model to forecast its behaviour (Chen & Huang, 2013).This model helps us to find a way to improve performance of RED algorithm by adjusting its parameters regarding the present and predicted future states of its queue.

The paper goes a follows: In section 2, we're going to discuss RED algorithm characteristics. We will introduce Markov model in section 3 and then explain various works in this scope in section 4. We will conclude in section 5.

2. RED (Random Early Drop)

RED is the most common AQM algorithm used widely in internet routers over the years. It uses average queue length calculated by exponential weighted moving averaging to detect congestion and determine present state of the queue. If the average is less than a determined minimum threshold, the algorithm decides that the queue is being underutilized and all incoming packets are allowed to access it. Between a maximum and minimum threshold value for average queue length is the region in which senders must be alert to avoid congestion, that's why some packets are dropped or marked randomly. This way they will be notified to reduce their sending rate. Number of packets dropped in this stage is determined by a parameter called packet drop probability. For every packet, this is the probability of being dropped or marked. For an average length more than the maximum threshold, all incoming packets must be dropped because we are considering the queue congested in that case.

The formula for dropping probability is as follows:

$$\begin{cases} p = 0 & , if\ q_{avg} < min_{th} \\ p = \dfrac{P_b}{1 - count * P_b} & , if\ min_{th} < q_{avg} < max_{th}\,. \\ p = 1 & , if\ q_{avg} > max_{th} \end{cases} \qquad (1)$$

Where, count is the number of packets since last dropped packet (Floyd and Jacobson, 1993) and P_b and q_{avg} are computed by equations (2)-(3).

$$P_b = max_p * \frac{(q_{avg} - min_{th})}{(max_{th} - min_{th})} \qquad (2)$$

$$q_{avg} = (1 - w_q) * q_{avg} + q * w_q \qquad (3)$$

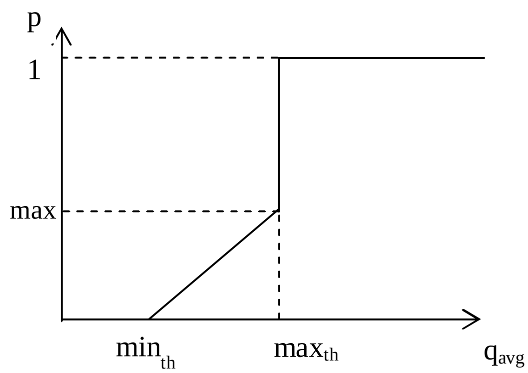
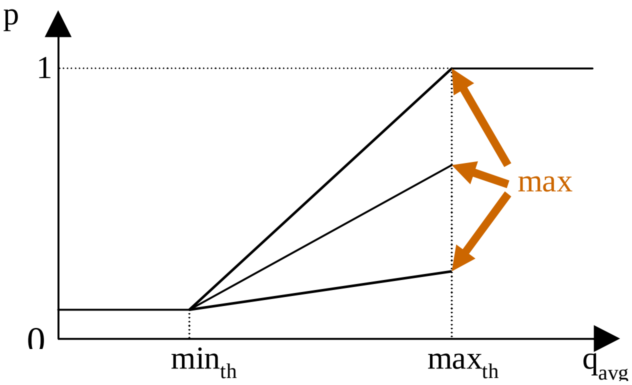


Figure 1- Drop Probability Function of RED

Many variants of original RED have been introduced over the years. Various works have used different approaches to deal with this problem. One of the most important ideas was adaptive versions of RED, in which, max_p changed with the state of the network. ARED, which was completed by Feng (2000) and Floyd (2001) was the first algorithm with the idea of changing maximum drop probability with queue state. FRED (Lin and Morris,1997) and SRED (Ott et al. 1999) are among other famous RED editions. Following is the algorithm developed and introduced as Feng's Adaptive RED or FARED (Feng et al. 2000):



FARED Adapts max_p based only on queue behaviour and not queue capacity. It increases max_p when q_{avg} crosses above max_{th}, decreases max_p when q_{ave} crosses below min_{th} and freezes max_p after changes to prevent oscillations.

3. Markov Model

Markov models are the fundamental building blocks upon which most of the quantitative analytical performance techniques are built. Markov models themselves are based on state space diagrams. Such diagrams are powerful descriptive tools. They are intuitive and natural, understandable by novices, yet rich enough to challenge experts. They can be applied across a wide range of applications. Markov models are often used to explain the current interactions between various system components. However, they can also be used for predictive purposes. Once a model is constructed, parameterized, and validated, it can be altered to predict what would happen if various aspects of the system's hardware or of the system's workload change. Thus, Markov models can be used for both descriptive and predictive purposes. To create such a model, the first step is to construct the state diagram by identifying all possible states that the modelled system may find itself. Second, the state connections (i.e., transitions) must be identified. Third, the model must be parameterized by specifying the length of time spent in each state once it is entered (or, equivalently, the probability of transitioning from one state to another within the next time period). After the model is constructed, it is "solved." This involves abstracting a set of linear "balance" equations from the state diagram and solving them for long term "steady state" probabilities of being in each system state. Once solved, the model can be validated and used for various performance prediction applications.

Within the context of Markov models, model construction consists of three steps: state space enumeration, state transition identification, and parameterization. State space enumeration involves specifying all reachable states that the system might enter. State transition identification indicates which states can be directly entered from any other given state. Parameterization involves making measurements and making assumptions of the original system. With Markov models, model construction involves identifying not only all the states in which the system may find itself, but also how long one typically stays in each state and which states are immediately accessible from any given state. Measurements, intuition, published results from the literature, and various assumptions are often used to parameterize a model.

Solving involves running an experiment on the newly constructed hardware and monitoring its performance. With simulation models, this involves running a software package (i.e. the simulator) and recording the emulated performance results. With analytical models, this involves solving a set of mathematical equations and interpreting the performance expressions correctly.

The baseline model can be used for the important purpose of prediction. One is much more interested in (and impressed by) a model that can predict what will happen before it actually does. By altering the baseline model (e.g., adding in future growth rate parameters, changing the hardware parameters to reflect anticipated upgrades) and then resolving the model, one can predict the future performance of a system prior to its occurrence (Menasce, 2004).

4. Markov Models use in congestion control

In this section, we will take a look at different works regarding modelling and modifying RED algorithm by means of Markov Models.

(Köhler et. al, 2000) has used a Markov model to analyze RED and assumes h TCP connections transmitting data over a RED queue. The model state is described by the random variables (Wi, Si, Ni), i∈ {1,…,h}and (A). The loss that a connection encounters is determined by$\mathrm{Li}(\mathrm{Wi}, \mathrm{A})$. All the TCP sources are saturated, therefore, they are waiting for $\sum_{i=1}^{h} Wi$ acknowledgements, i. e., the number of packets at the server $Q = \sum_{i=1}^{h}(Wi - \mathrm{Li}(\mathrm{Wi}, \mathrm{A}))$is the difference of the window sizes Wi and the lost packets Li(Wi, A). To build a discrete model, the queue can be observed at the same instants because within this time, for each TCP source exactly one round completes. The coupling consists of propagating the loss induced by the loss function p(A)) of the RED queue as feedback to the TCP connections to control their window size. The traffic arriving at the RED queue is the sum of all congestion

Vol.1 Issue.8, Pg: 62-72
November 2013

window sizes $\sum_{i=1}^{h} Wi$ representing the number of sent packets during the next round of each connection. The number of lost packets within the next round is the sum of lost packets of all TCP connections$\sum_{i=1}^{h}$ Li(A, Wi).

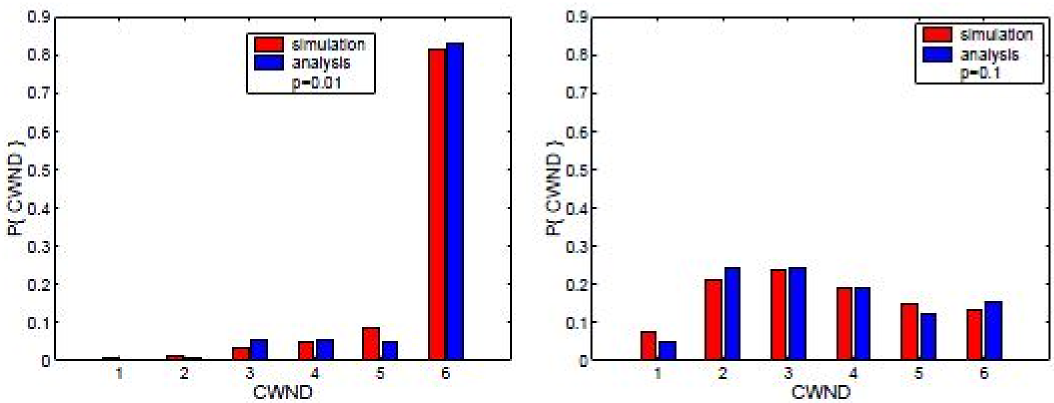


Figure 3- Comparison of analytical and simulated TCP congestion window distribution

Figure 3 presents a comparison of the analytically derived distribution with simulation results generated with the NS simulator. The distribution of the congestion window CWND for drop probabilities p=0.1 1 and p=0.01 are found in good accordance with the simulated results. The results of the study show that proper dimensioning of the RED queue parameters is crucial to achieve a benefit and not to experience drawbacks from the RED queue (Köhler et. al, 2000).

Using the same approximations, (Wurtzler, 2002) claims that WRED can be analyzed in a very similar way to RED. WRED uses the same parameters as RED, but it has the ability to perform RED on traffic classes individually. Figure4 shows a WRED system in which both classes have the same max_p but different min_{th}.

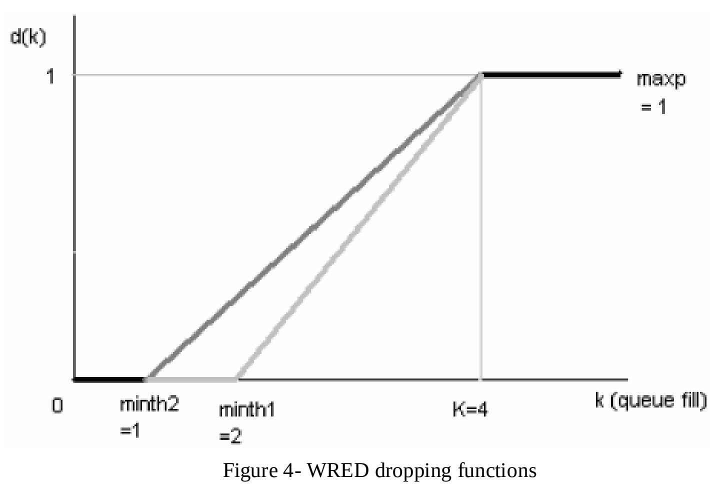


Figure 4- WRED dropping functions

WRED analysis is similar to RED analysis, but the transition rates for the CTMC will differ and the general complexity of the Markov chain will increase. Figure 5 shows the system model proposed for WRED.

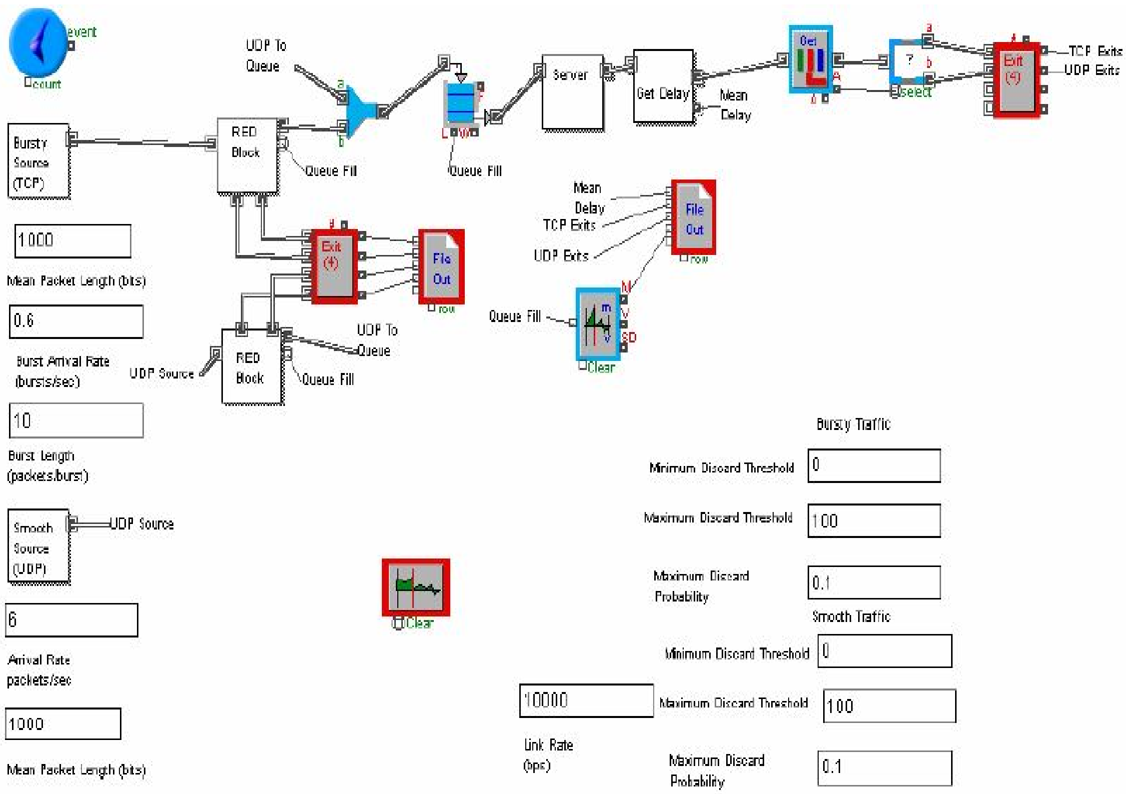


Figure 5- System model for WRED

Through an iterative process, WRED parameters were found that provide close to an exact match in dropping probability for two classes with different arrival statistics. Many of the benefits of RED were shown to be true for proposed WRED model, and the cost in overall dropping probability, as compared to a tail drop queue, was relatively low(Wurtzler, 2002).

(Barakat& Altman, 2000) has applied Markovian fluid model to model TCP deploying RIO and RED algorithm and has claims that in the under-subscription case, the RIO scheme is more biased against large reservation connections and long RTT ones than the other schemes. This is not the case in the over-subscription case where it provides a better performance.

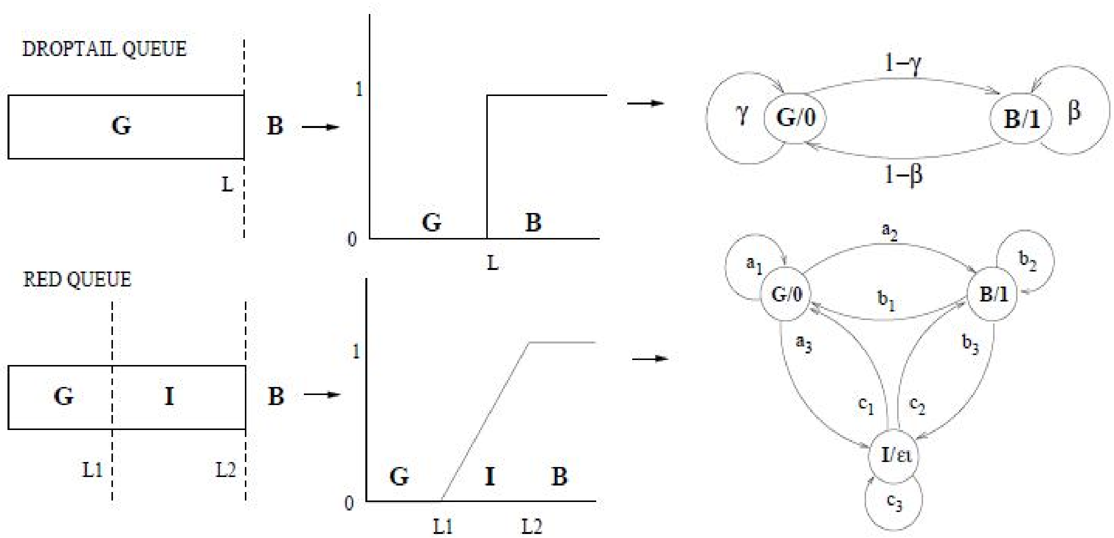
A Markov-fluid queue is a special type of queue, where the buffer content, the amount of work in the system, changes at a constant rate. This constant rate changes after a random period of time. The buffer content cannot be negative, so when the buffer content decreases to zero, the buffer content stays zero as long as the constant rate is negative (van de Leur, 2008).

(Suthaharan, 2004) proposes a solution to this problem using Markov chain based decision rule. They model the oscillation of the average queue size as a homogeneous Markov chain with three states and simulated the system using the network simulator software NS-2. They claim based on simulations that their proposed scheme successfully estimates the maximum packet dropping probability for Random Early Detection, detects the congestion very early and adjusts the packet-dropping probability so that RED can make wise packet-dropping decisions. The proposed scheme is also said to provide improved connection throughput and reduced packet loss rate.

(Singh et. al, 2005) present an analytical study targeted at statistically capturing the loss behaviour of a RED queue. They utilize a finite-state Markov chain model. Starting from recursive equations of the model, they derive Equivalent closed-form equations and numerically validate the matching of recursive and closed-form equations. Further, they apply it to monitor the average RED queue size in a number of sample topologies illustrating their practicality. They argue that their model can adapt to the changing network conditions. The model is shown in figure 6.

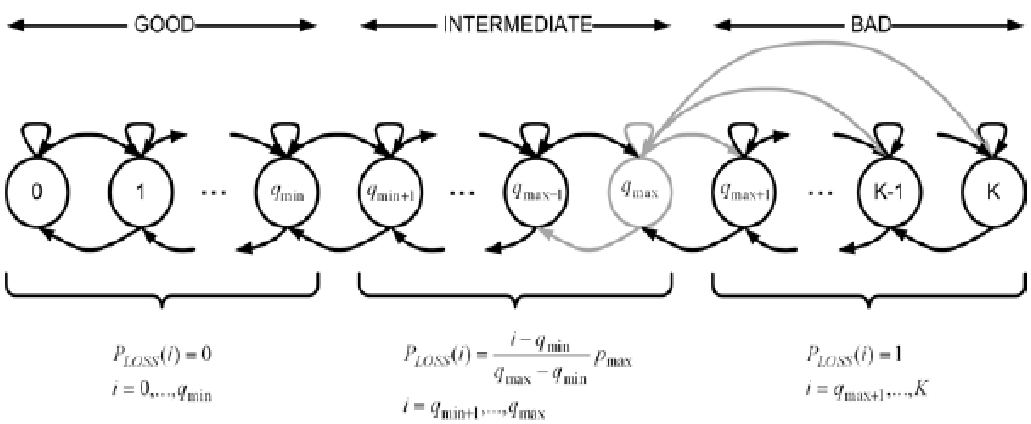
(Sharma et. al, 2002) consider a finite queue with its arrivals controlled by the Random Early Detection (RED) algorithm. This is one of the most prominent congestion avoidance schemes in the Internet routers. The aggregate arrival stream from the population of TCP sources is locally considered stationary renewal or MMPP with general packet length distribution. They have claimed their model to be accurate with the help of simulations.

Figure 6- Markov modeling of a drop-tail queue and a RED queue with max_p=1



(Bauer et. al, 2007) propose an analytical model to capture the dynamics of the RED algorithm. They develop a system of recursive equations that describes the packet dropping behavior of the RED algorithm. Using a notion from the theory of random walks, they derive an exact-closed form expression that characterizes the loss characteristics of a RED queue. They have validated the derived formula by a numerical comparison with the recursive equations.

Figure 7- M/D/1/K approximation of the steady-state behavior of RED under quasi-stationary assumptions



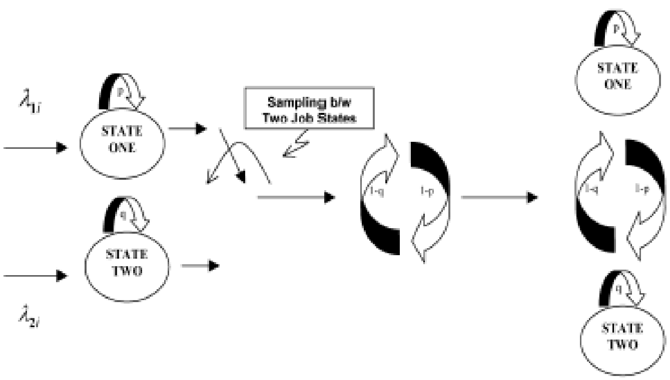
In (Zhenzhen et. al, 2010) a novel closed-loop feedback TCP/AQM (Transfer Control Protocol/Active Queue Management)model is proposed using a discrete-time Markov chain, and a way to calculate the equilibrium distribution of this model is given. In the model, system time is divided into time slots, the bottleneck router queue model and TCP window size model in each slot are analyzed. By combining adjacent slots, an integrated TCP/AQM analytical model is developed, the average values of packets dropping ratio and queue length in the router and TCP sending rate are estimated. The proposed TCP/AQM model is extended to a TCPUDP (Control User Datagram Protocol)/AQM model, to analyze the TCP/AQM system performance when UDP flows exist. By implementing this model on Matlab, the validity of the model to analyze the closed-loop feedback TCP/RED (Random Early Detection) system has been verified. Figure 8 is the transition matrix for this model.

Figure 8- Transition matrix for proposed model in (Zhenzehn et. al 2010)

$$\boldsymbol{P} = \begin{bmatrix} 1-\lambda(k)(1-f(0))\delta t & \lambda(k)(1-f(0))\delta t & 0 & \cdots & \cdots & 0 \\ \mu\delta t & 1-\left(\lambda(k)(1-f(1))+\mu\right)\delta t & \lambda(k)(1-f(1))\delta t & \cdots & \cdots & 0 \\ 0 & \mu\delta t & 1-\left(\lambda(k)(1-f(2))+\mu\right)\delta t & \cdots & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & \mu\delta t & 1-\mu\delta t \end{bmatrix}$$

(Yar et. al, 2012) proposes a threshold based analytical model based on standard RED mechanism. Based on the Markov Modulated Poisson Process (MMPP), which has been extensively used to model B-ISDN sources, such as voice and video, as well as characterizing the superposed traffic. It captures the burstiness and correlation properties of the network traffic. In addition to characterizing the desired properties of B-ISDN applications, these models are analytically tractable and produce results that are acceptable approximations to reality. Sampling concept for their proposed model is given in figure 9.

Figure 9-Sampling concept for proposed model in (Yar et. al, 2012)



5. Conclusions and future work

In this paper we have reviewed some works on modelling RED and deploying Markov model to optimize RED performance or study its behaviour to develop suitable modifications to get a better algorithm with higher performance. Yet, the use of Markov model to predict and deploy this prediction in modifying RED parameters remain for more extensive studies to be used in practice. This deployment can include tuning of different RED parameters to get a more stable system.

REFERENCES

AweyaJ, OuelletteM, andMontunoD. Y, 2001, A Control Theoretic Approach to Active Queue Management, Computer Networks Journal, vol. 36, no. 2–3, pp. 203–235.

Barakat C, Altman E, 2000, Quality A Markovian Model for TCP Analysis in a Differentiated Services Networkof Future Internet Services, Springer, Lecture Notes in Computer Science, Volume 1922, pp. 55-67.

Bauer C, Yousefi'zadeh H, Jafarkhani H, 2007, Global Telecommunications Conference, IEEE GLOBECOM '07, pp.1818 – 1822.

BradenB,ClarkD, CrowcroftJ, D. B. Deering, S,Estrin D,FloydS,JacobsonY,PartridgeG.M.C, PertersonL,RamakrishnanK,ShenkerS, WroclawskiJ, and.ZhangL, 1998, Recommendationson Queue Management And Congestion Avoidance In The Internet, RFC 2309.

Chen W, Yang S.H, 2009, The Mechanism Of Adapting RED Parameters to TCP Traffic, Computer Communication 32, pp. 1525-1530.

FengW,KandlurD.D,SahaD, ShinK.G, 2000, A Self-Configuring RED Gateway, In IEEE INFOCOM2000,Tel-Aviv, Israel, March 26-30.

FiroiuV.and Borden M. 2000, A Study of Active Queue Management for Congestion Control, In IEEE INFOCOM2000, Tel-Aviv, Israel, March 26-30.

Floyd S,JacobsonV, 1993, Random Early DetectionGateways for Congestion Avoidance, ACM/IEEETransaction onNetworking, vol.I, no. 4, pp. 397-413.

Floyd S, Gummadi R, Shenker S, 2001,Adaptive RED: An Algorithm for Increasing the Robustness of RED. Technical Report.

HeinanenJ,Baker, F,WeissW, andWroclawskiJ,1999,Assured Forwarding PHS, RFC2579.

Jacobson V, Nochols K, Poduri K and Cisco Systems, 1999, Technical Report.

Köhler S, Menth M, Vicari N, 2000, Analytic Performance Evaluation of the RED Algorithm for QoS in TCP/IP Networks, Department of Distributed Systems Institute of Computer Science University of Würzburg, Germany, Report No. 259.

LinD, MorrisR, 1997, Dynamics of Random Early Detection.In proceedings of ACM SIGCOMM.

LowS. H, PaganiniF, WangJ, and. DoyleJ. C, 2003, Linear Stability Of TCP/RED and a Scalable Control,Computer Networks Journal, vol. 43, no. 5, pp. 633–647.

NS-2, Network Simulator Documentationhttp://www.isi.edu/nsnam/ns/S.

Ott T. J, LakshmanT. V,Wong,1999, SRED: stable RED. In Proceedings of IEEEINFOCOM,.

ParrisM.JeffayK, and SmithF.D 1999, Lightweight Active Router Queue Management for Multimedia Networking, In Proceedings of SPIE, pages 162-174, San Jose, CA, USA, Jan 25-27.

Sarkar S, SikkaG, KumarA,2012, Classification of Active Queue Management Algorithms for Application Specific Traffics, International Journal of Advanced Research in Computer Science and Software Engineering,Volume 2, Issue 5, pp. 115-124.

Sharma, V, Virtamo, J and Lassila P, 2002,Performance analysis of the random early detection algorithm, Probability in the Engineering and Informational Sciences, 16 (3), pp. 367-388.

Singh M. B, Yousefi'zadeh H, Jafarkhani H, 2005, A Finite-State Markov Chain Model for Statistical Loss Across a RED Queue, IEEE Systems Communications, 2005 Proceedings.

Suter B, LakshmanT.Y,StiliadisD and ChoudhuryA.K.1999, Buffer Management Schemes for Supporting TCP In Gigabit Routers with Per-Flow Queuing. IEEE Journal on Selected Areas in Communications, vol. 17, no. 6, pp. 1159-1169, June.

Suthaharan S, 2004, Grey Model Based Congestion Control for TCP. 37th Annual Simulation Symposium, Hyatt Regency Crystal City, Arlington,VA.

TanL, ZhangW, PengG, ChenG, August 2006, Stability of TCP/RED Systems in AQM Routers, IEEE Transactions On Automatic Control, vol. 51, No. 8.

Van de Leur, Markov-Fluid Queues, Bachelor Thesis,2008, Korteweg de Vries Institute for Mathematics, Faculty of Science, University of Amsterdam.

Wurtzler M,2002, Analysis and Simulation of Weighted Random Early Detection (WRED) Queues, MSc. Thesis, Northwestern University, Evanston, Illinois.

YarAsfand,.Awan I.Uand.WoodwardM.E, 2012, Performance Modeling Of A MultipleThreshold Red Mechanism For Burstyand Correlated Internet Traffic WithMMPP Arrival Process,, I.J. of Simulation, Vol. 7, No. 1, pp. 23-36.

Zhenzhen C, Yang X, Caixia C, 2010, A Novel Markov Analytical Model For The TCP/AQMSystems, Vol.27, No.1, Journal Of Electronics, China.