A SURVEY ON RED QUEUE MECHANISM FOR REDUCE CONGESTION IN WIRELESS NETWORK

Kachhad Krishna Chandulal

Department of Computer Engineering, Marwadi Education Foundation – Faculty of PG Studies & Res. in Engg. & Tec. Rajkot, Gujarat, India

---***---

Abstract – *Now a day's use of wireless networks for access the internet is increasing. It shows that use of the main transport layer protocol TCP will be increase in near future. The number of applications running over computer networks has been increasing tremendously, which ultimately results in congestion. Congestion increases both delay and packet loss. Network congestion can be controlled by several methods, such as Random Early Detection (RED) which is the most well known and widely used queue mechanism to avoid congestion. Analysis of various RED mechanism base on that Fuzzy Logic RED (FLRED) uses fuzzy logic to avoid linearity and parameterization problem in RED and Early Congestion Control (ECC) mechanism is dynamically changes the value of the window field from TCP header where as Explicit Non-Congestion Notification (ENCN) is notifying non-congestion in the router and Three section Random Early Detection (TRED) is divided into three section light, moderate and high load for calculate packet dropping probability also the Efficient LAL Random Early Detection (ELALRED) is founded on the principles of the RED mechanism with LAL philosophy but the Hemi-Rise Cloud Model (CRED) was proposed non linear packet loss strategy was used. Based on observation CRED effectively controls the oscillation of the average queue length.*

***Key Words*: Congestion control, TCP, Active queue management (AQM), Random early detection (RED), Hemi-rise cloud (CRED), Fuzzy logic random early detection (FLRED), Early congestion control (ECC), Explicit non-congestion notification (ENCN), Nonlinear, Three section random early detection (TRED).**

1. INTRODUCTION

Wireless network is the transfer of information between two or more points that are not connected physically. A wireless network is a flexible data communications system, which uses wireless media such as radio frequency technology to transmit and receive data over the air. Wireless means transmitting signals using radio waves instead of wires. Wireless networks use electromagnetic waves to communicate information from one point to another without any physical connection. The Transmission Control Protocol (TCP) is a transport layer protocol that provides reliable data transfer, connection oriented service, flow control and congestion control [3]. Transmission control protocol is mostly used protocol in current communication network and over the internet. TCP also controls the rate of transmission from the sender node across the network with end to end feedback, through packet. The protocol supports reliable data transport by establishing a connection between the transmitting and receiving ends. The TCP protocol attains congestion control through an end-to-end algorithm which computes the appropriate sender congestion window by means of the estimated traffic conditions [4]. Nowadays, the rapid growing of network technology and applications, network congestion has been increasingly serious, and the congestion control is becoming more and more urgent. Congestion control strategy based on intermediate node (router) have been proposed and gotten attention to compensate for the lack of the source end TCP [1]. Results of congestion include a high delay, wasted resources, and even global synchronization. The aim then would be to control congestion or, more ideally, avoid congestion. Congestion control techniques have been developed called Active Queue Management (AQM) [3-5]. A congestion control scheme based on active queue management has become a research hot spot in the industry.

1.1 Active Queue Management (AQM)

Congestion in Internet occurs when the link bandwidth exceeds the capacity of available routers which results in buffer bloat problem. This results in long delay in data delivery and wasting of resources due to lost or dropped packets. To resolve these mentioned congestion problems two approaches are identified. The most basic AQM approach is the Drop Tail (DT) scheme where packets arriving in a queue are dropped with probability one when the queue is full. However, DT can cause that all TCP flows through the congested queue reduce their transmission rates at the same time, because each TCP source reduces its window every time it detects a packet loss. Also, in each congestion episode, there is a high chance of dropping a packet from each active flow. This phenomenon is known as global synchronization [7]. Drop Tail queue suffer the problem of global synchronization in which queue is over utilized and underutilized at alternative period of time.

To overcome such inconveniences several other AQM mechanisms have been proposed, in which instead of waiting for the queue to be full to act they respond in advance to congestion by dropping or marking packets. Accordingly, Floyd and Jacobson [7] proposed an algorithm called Random Early Detection (RED) which basically detects congestion by estimating the average occupation of the queue. The RED algorithm functions by detecting incipient congestion and notifying the Transmission Control Protocol (TCP) by probabilistically dropping packets before the queue when a router fills up. Briefly, the algorithm works by maintaining an average queue size. As the average queue size varies between the minimum (minth) and maximum (maxth) thresholds, the packet dropping probability linearly changes between zero and maximum drop probability Pmax. Thus, the packet dropping probability function is linear to the change of the average queue size (ave). If the average queue size exceeds the maximum threshold, all arriving packets are dropped. Since the packet dropping mechanism is based on the moving average algorithm, RED can control the transient congestion by absorbing arrival rate fluctuations. Although RED is a significant improvement over simple Drop Tail that simply drops all incoming packets when a Drop Tail queue is full, RED is particularly sensitive to the traffic load and the parameters of the scheme itself [5].

RED algorithm forecasts buffer queue length first and takes measures according to values predicted in advance such as adjusting sending rate, which can avoid network congestion to a certain extent and effectively solve the problem of the global synchronization [1].

1.2 Overview of RED (Random Early Detection)

Floyd and Jacobson [7] proposed an algorithm called Random Early Detection (RED) which basically detects congestion by estimating the average occupation of the queue. RED also known as Random early drop and Random early discard. RED algorithm solve the problem of the global synchronization but RED is sensitive to parameters and traffic load which exists uncertainty. RED is highly sensitive to its parameter settings namely minimum threshold (maxth), minimum threshold (minth), maximum packet dropping probability (Pmax) and weighting factor (Wq) [1].

RED uses a mechanism early detection of packet drop without waiting to queue overflow. When congestion will happen, router discards the arriving packets with certain probability. This can inform the sender to adjust size of sending window before congestion happen. RED gets the average queue length of router to predict the network congestion. RED algorithm maintains two parameters maximum threshold and minimum threshold. RED uses the weighted average function to predict the average queue length and compares the results with the minimum and maximum threshold predefined. If it is less than the minimum threshold, the arrival packet will be forwarded normally; if estimated result is greater than the maximum threshold, all the arriving groups will be discarded otherwise the packet reaching will be randomly discarded or tagged according to certain probability [1].

The RED scheme drops packets with a probability by computing the average queue length (ave) to notify traffic sources about early stage of network congestion. Where Wq is the weighting factor, ave is the average of the queue length. In addition, the average queue length is expressed as

$$ave = (1 - Wq)ave + Wq\ q \qquad (1)$$

In the aforementioned formula, q is the instantaneous queue length, and $Wq \in [0, 1]$ is the weighting factor.

RED has three more parameters, i.e., minimum threshold minth, maximum threshold maxth, and the maximum dropping probability Pmax at maxth. If the average queue length is below minth, RED drops no packets. However, if the average queue length increases above minth but is below maxth, RED drops incoming packets with a probability proportional to the average queue length linearly. When the average queue length exceeds maxth, all the arriving packets are dropped. The packet dropping probability (Pb) being the function of the average queue length is calculated by

$$P_b = \begin{cases} 0, & ave \in [0, minth] \\ maxp(ave-minth\ /\ maxth-minth), & ave \in [minth, maxth] \\ 1, & ave \in [maxth, +\infty] \end{cases} \qquad (2)$$

RED is an improvement over simple Drop Tail. RED avoids the TCP starvation problem and global synchronization but compared with Drop Tail, RED exhibits a lower delay and a higher throughput and packet loss [2].

2. LITERATURE SURVEY

The Hemi-Rise Cloud Model (CRED) [1] was proposed a new RED algorithm for congestion control and used non linear packet loss strategy. RED algorithm avoids the global synchronization but RED performance is sensitive to traffic load and parameter means randomness and fuzziness. Hemi-rise is typical clouds model that solving fuzziness and randomness of RED algorithm. In CRED combine drop tail algorithm with RED algorithm. Packet dropping strategy of CRED is described below.

(1)Average queue length is calculated for each packet arriving.
(2)If queue is not empty then

$$Lavg = (1 - Wq) * Lavg + Lnow * Wq \qquad (3)$$

Where Lnow is the current length of router queue, Lavg is the average of the queue length and Wq is constant. If Lavg is below the Lmin then packet is not drop. If Lavg increases above Lmin but is below Lbuffer, CRED drops incoming packets with a probability of average queue length. According to CRED Lbuffer is the Lmax.

$$P = Ptmp\ /\ (1 - count * Ptmp) \qquad (4)$$

Where Ptmp is the transitional drop loss rate. When Lavg exceeds Lbuffer, all arriving packets are dropped. CRED algorithms proposed effectively improve problems of parameters and load sensitivity in RED and ARED algorithm. CRED algorithm is also good for stability of the network.

Advantages: CRED algorithm is improve the stability and get better performance than the RED. Also improve the uncertainty of parameters. CRED effectively controls the oscillation of the average queue length and stably operates under various network environments. Quality of service is greatly improved such as robustness and stability.

Disadvantages: It is complex queue mechanism because necessary the cloud model properties Ex (expectation value), En (entropy) and He (ultra-entropy) for finding dropping probability.

The Fuzzy Logic RED (FLRED) [2] was proposed a new congestion control method which extends RED. FLRED uses fuzzy logic to avoid the linearity and parameterization problem in RED. FLRED uses two congestion indicators that are aql (average queue length) and Dspec (delay speculation). At a time only one indicator is good, either aql is good or Dspec is good. These two indicators are used to calculate the Dp (dropping probability) for incoming packets within a Fuzzy Inference Process (FIP). In FIP based method, FLRED is implemented in four sequential steps that are fuzzification, rule evaluation, aggregation and defuzzification. The first step of the process is fuzzification, in which triangular is used to formulate the input and output linguistic variables. The second step of the FIP process is rule evaluation, which evaluates the rules. The third step of the FIP process is aggregation, the probabilities of the output linguistic terms obtained by the applied rules. The last step of the FIP process is defuzzification, in which the output linguistic variable values are produced based on fuzzy set. For defuzzification employed Center of Gravity (COG). In short for finding Dp get one common point through aql and Dspec indicators.

Advantages: FLRED queue mechanism decreasing both delay up to 1.5 to 4.5% and packet loss up to 6 to 30% under heavy congestion than RED and effective RED (ERED).

Disadvantages: Under light congestion, increase delay than RED and ERED.

The Explicit Congestion Control (ECC) [3] presents a new approach to improve the performance of TCP in ad hoc networks. The ECC is based on adjusting the window field of TCP segments. The ECC mechanism is dynamically changes the value of the window field from TCP headers according to the utilization of the router queue. When queue is higher than the threshold Q*l the value of the window field of the TCP header is reduced to the percentage (Q – q / Q) of bytes available in the queue. If queue is not higher than the threshold Q*l the value of the window field of the TCP header is same.

$$w = Q - q / Q * w \qquad q > Q*l \qquad (5)$$

ECC operates in the same way as drop tail but one characteristic is added to reduce the value of the window field of the TCP header.

Advantages: In ECC approach the good put is 8.28% higher than the Drop Tail and RED. Also ECC reduce the packet loss. Comparing ECC approach is reduce an average delay 28.40% less than Drop Tail and 36.21% smaller than RED.

Disadvantages: ECC approach only suitable for ad-hoc networks and grid topology.

The Explicit Non-Congestion Notification (ENCN) [4] presents a new approach for Active Queue Management (AQM) technique. Most of works are based on congestion notification means Explicit Congestion Notification (ECN) but the absence of congestion notifications still not research. The ENCN approach overcome unwanted empty queue phenomenon and take advantage of the non-congestion state of the queue. The TCP action with AQM action, every time transmitter receives a marked ACK and it will modify its congestion window. If instantaneous queue length (Qinst) is lower than a threshold (Qmin), router mark with probability one (ENCN = 1). At transmitter side if TCP is in slowstart, it will calculate its new congestion window. If TCP is in congestion avoidance the sender will increase (W + 1) its congestion window by one packet size per round trip time(RTT). If instantaneous queue length (Qinst) is greater than or equal to a threshold (Qmin), router is unmark (ENCN = 0) that time transmitter side if TCP is in slowstart it will continue in slowstart and if TCP is in congestion avoidance the sender will decrease (W – 1) its congestion window by one packet size.

Advantages: ENCN approach gives better throughput compared with other AQM techniques because empty queue phenomenon causes performance loss in terms of throughput and ENCN overcome the empty queue phenomenon.

Disadvantages: ENCN approach gives better throughput only for TCP Reno and TCP sack variants than other variants.

The Three section Random Early Detection (TRED) [5] based on nonlinear RED presents a minimal adjustment to RED. TRED aimed at solving RED's link underutilization and large delay in low and high traffic load scenario problems. In TRED average queue length between two thresholds is divided into three sections: light, moderate and high load. If average queue size between the minth and minth + Δ, allow the incoming packets with low load, where Δ = (maxth – minth)/3. If average queue size between the minth + Δ and minth + 2Δ, allow the incoming packets with moderate load and average queue size between the minth + 2Δ and maxth, allow the incoming packets with high load. TRED find the packet dropping probability based on curve. Based on curve formula finally get the packet dropping probability. Through the simulation the performance of TRED solves RED's low bandwidth utilization in low load scenario and large delay in high load scenario.

Advantages: Very little work needs to be done to migrate from RED to TRED. Also TRED queue mechanisms reduce delay 4.8ms at high load compare to RED.

Disadvantages: TRED queue mechanism not used Explicit Congestion Notification (ECN).

The Efficient Learning Automata Like Random Early Detection (ELALRED) [6] presents a algorithm the concept of a Learning Automata Like (LAL) mechanism devised for congestion avoidance in wired networks. ELALRED is the operations of the existing RED congestion avoidance mechanisms, augmented with an LAL philosophy. Total five action used in ELALRED algorithm: Forced_drop, Minimum_exceed, Unforced_drop, No_drop and Maximum_exceed. Forced_drop is chosen when the average queue size is above the maximum threshold (avg > maxth). Minimum_exceed is chosen when the average queue size exceeds the minimum threshold and average just crosses minth (minth < avg < maxth). Unforced_drop is chosen when the average queue size lies between the minimum threshold and maximum threshold (minth < avg < maxth). No_drop is chosen when the average queue size lies below the minimum threshold (avg < minth). Maximum_exceed is chosen whem the average queue size exceeds the maximum threshold and average just crosses maxth (avg <= maxth). ELALRED maintain a maximum likelihood estimate of how profitable action has been and this is inferred by examining the estimate vector d(t). The algorithm also maintain and update probability vector p(t). The ELALRED mechanisms avoid congestion in the network and maximize the number of packets.

Advantages: ELALRED algorithm reduces the packet drops at the gateways compared to the LALRED and RED. Using ELALRED more packets are acknowledged to the sender. Also ELALRED scheme is superior to LALRED and RED because suitable even networks with 100 nodes.

Disadvantages: ELALRED scheme is avoid congestion for wired networks.

Table -1: Comparison of Various Queue Mechanisms

Queue Name	Dropping Probability	TCP Variants	Queue Size	ECN Notification	Complexity
CRED[1]	Yes	TCP	300 Packet	Yes	Yes
FLRED[2]	Yes	TCP	20 Packet	No	Yes
ECC[3]	No	TCP NewReno	97KB	No	No
ENCN[4]	No	TCP NewReno & TCP Sack	17 Packet	Yes	No
TRED[5]	Yes	TCP NewReno	120 Packet	No	Yes
ELALRED[6]	Yes	TCP	100 & 1000 Packet	No	Yes

3. CONCLUSIONS

The study and research of queue mechanism for congestion control continues to be an active area for the researchers. In this survey paper, first studied about the Random Early Detection (RED) queue mechanism and their many variants. After that surveyed the earlier queue management techniques such as Drop Tail and RED. Then, came to the active queue management techniques such as CRED, FLRED, ECC, ENCN, TRED and ELALRED mechanism. This paper gives an idea for various variants of RED queue mechanism and various dropping probability for controlling congestion and each has its own limitations.

Many study and experimental results show that CRED gives better performance in comparison with other techniques such as FLRED, ECC, ENCN, TRED and ELALRED because it maintains stable queue length. Overall CRED effectively controls the oscillation of the average queue length and stably operates under various network environments also quality of service is greatly improved such as robustness and stability.

ACKNOWLEDGEMENT

I acknowledge to Marwadi Education Foundation, my parents, and journal team for their support in my research work.

REFERENCES

[1] Y. Zhao, Z. Ma, X. Zheng and X. Tu, "An Improved Algorithm of Nonlinear RED Based on Membership Cloud Theory," in Chinese Journal of Electronics*,* vol. 26, no. 3, pp. 537-543, 5 2017.

[2] Abualhaj, Mosleh M., Ahmad Adel Abu-Shareha, and Mayy M. Al-Tahrawi. "FLRED: an efficient fuzzy logic based network congestion control method." Neural Computing and Applications: 1-11.

[3] M. Talau, M. Fonseca, A. Munaretto and E. C. G. Wille, "Early congestion control: A new approach to improve the performance of TCP in ad hoc networks," *2016 7th* International Conference on the Network of the Future (NOF)*, Buzios, 2016, pp.* 1-6.

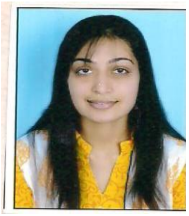
[4] L. M. A. Sup, R. M. de Moraes and A. Bauchspiess, "Explicit non-congestion notification: A new AQM approach for TCP networks," *2017 13th* International Wireless Communications and Mobile Computing Conference (IWCMC)*,* Valencia, 2017, pp. 1239-1244*.*

[5] C. W. Feng, L. F. Huang, C. Xu and Y. C. Chang, "Congestion Control Scheme Performance Analysis Based on Nonlinear RED," in IEEE Systems Journal*,* vol. PP, no. 99, pp. 1-8.

[6] S. G. Mahajan, "Efficient LALRED for Congestion Avoidance Using Automata-Like Solution," *2015* International Conference on Emerging Information Technology and Engineering Solutions*,* Pune, 2015, pp. 1-6.

[7] S. Floyd and V. Jacobson, "Random early detection gateways for congestion avoidance," in IEEE/ACM Transactions on Networking, vol. 1, no. 4, pp. 397-413, Aug 1993.

BIOGRAPHIES



Krishna Kachhad is currently a second-year M.E. (Computer) student at Marwadi Education Foundation –Faculty of PG Studies & Res. in Engg. & Tec. Rajkot, Gujarat, India. Her research interests primarily include Congestion control in wireless network.