International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 06 Issue: 08 | Aug 2019 www.irjet.net p-ISSN: 2395-0072

Survey on Different Types of Congestion Control Algorithms

G. Alamelu Mangai

Department of Computer Science, Bharathidasan University, Tamil Nadu, India

---***---

Abstract - *Congestion is one of the main issues in wireless network. Internet is considered as the worldwide system of many computers and their networks which are linked together by wired or wireless media. The growth of internet demands effective congestion control mechanism to be converted into successful. In this paper discussed various congestion control algorithms and the methods using the algorithms.*

Key Words:- Congestion control, Wireless Network, RED (Random Early Detection), Fuzzy, SNR Ratio

1. Introduction

Congestion in a network is mobility by delay and pocket loss in the network. Wireless network are inherently limited by battery power and bandwidth constraints. They are characterized by mobility, random changes in connectivity, fluctuations in channel and interference duo to neighboring nodes etc. due to these factors packet loss of wireless network is much more than that of a wired network, in which packet loss occurs mainly due to congestion.

2. Congestion Control

Congestion is a network may occur if the load on the network is greater than the capacity. In other word Congestion is characterized by delay and loss of packets in delivery. Congestion Control refers to the mechanisms and techniques that can either prevent congestion before it happens or remove after it happens.

3. Congestion Control Algorithms

3.1 URED (Upper Random Early Detection)

In this algorithm is presents the minimal changes of the basic RED algorithm. URED is mainly proposed the efficient congestion control algorithm to design denial of service attack. In this algorithm we have introduced new threshold (Upper threshold) for better use of buffer space, to queue more packets which reduces packet drops due to constant packet drop probability when average queue size is greater than max.

RED and other enhanced RED algorithm increases linearly up to packet dropping probability in maximum. If average queue size goes greater than maximum.

In the URED algorithm used three threshold minimum threshold, maximum threshold and upper threshold and it will also increase adaptability of RED.

Pseudo code of the URED algorithm with upper threshold

Initialization:

avg = 0; count = - 1

For each packet arrival

Calculate the new average queue size avg;

if the queue is non empty

$avg = (1 - w_q) *avg + w_q * q$

else

m = f(time – q_time)

avg = $(1- w_q)^m$ * avg

if $Min_{th} \le avg \le Max_{th}$

increment count

calculate probability Pa

$P_b = Max_p(avg – Min_{th})/(Max_{th} – Min_{th})$

$P_a = P_b(1- count *P_b)$

With Probability Pa:

Mark the arriving Packet

Count =0

else if $Max_{th} \le avg < U_{th}$

Increment count

Calculate probability P_a

$P_b = (1 – Max_p)*((avg – Max_{th})/(U_{th} – Max_{th}))$

$P_a = P_b(1- count *P_b)$

With Probability P_a:

Mark the arriving Packet

count = 0

else if $U_{th} \le avg < BS$

Drop the packet

count =0

else

count = - 1

When queue becomes empty

q_time = time

Saved Variables:

avg: average queue size

q_time: start of the queue ideal time

count: packets since last marked packet

BS: Buffer size

Fixed Parameters:

wq : queue weight

Min_{th} : Minimum Threshold

Max_{th} : Maximum Threshold < Half of the BS

U_{th} : Upper Threshold < $3/4^{th}$ of the BS

Max_p : Maximum Value for P_b

P_a : Current packet marking probability

q : Instantaneous queue size

t : time

f(t) : Linear function of time t

3.2 Multipath Congestion Control

Multipath congestion control aims to efficiently allocate resources inn network where connection span multiple paths. The starting point of all practical multipath congestion control algorithms is the theoretical work by Kelly and voice that have proposed a class of congestion control algorithms where senders regulate the transmission rates based only on loss information received from the network.

Several multipath congestion control algorithms has been proposed for wireless network. They can be classified into two approaches: Uncoupled congestion control, and Coupled congestion control approach.

Design of Multipath TCP Veno

TCP-Veno is combination of TCP Vegas and TCP Reno to take advantage of distinguishing pocket losses caused by random error of wireless links or by network congestion. In the first algorithm is described its congestion window update rules and then the second algorithm described a sub flow on path r adjusts its congestion window size.

Algorithm 1 Single-path TCP Veno Background

Increase:

if n_ backlog < β then

/* Whenever receiving a new ACK */

w ← w + 1/w

else

/* Whenever receiving two new ACKs */

w ← w + 1/w

end if

Decrease:

if n backlog < β then

/* Packet loss was caused by random error */

$w \leftarrow w - w/5$

else

/* Packet loss was caused by network congestion */

$w \leftarrow w - w/2$

end if

Algorithm 2 Multipath TCP Veno algorithm for the sender on

path *r*.

Increase:

if n_ backlog$_r$ < max(1, θ_r β) then

/* Whenever receiving a new ACK */

$w_r \leftarrow w_r + a_r/w_r + \alpha_r/w_r$

else

/* Whenever receiving two new ACKs */

$w_r \leftarrow w_r + a_r/w_r + \alpha_r/w_r$

end if

Decrease:

if n_ backlog$_r$ < max(1, θ_r β) then

/* Packet loss was caused by random error */

$w_r \leftarrow w_r - w_r/5$

else

/* Packet loss was caused by network congestion */

$w_r \leftarrow w_r - w_r/2$

end if

3.3 Hop-by-Hop Congestion Control

In this algorithm sometimes also called backpressure each router (hop) along end-to-end path sends feedback to the directly preceding router, which executes control mechanisms based on this feedback. In general, the goal of hop-by-hop schemes was to obtain reliability inside the network, for example, by buffering packets at each hop until they can be sent on (because-there is no more congestion) and only transmitting as much as the transfer and high reliability.

Algorithm pseudo code

```
Assume Queue as Circular Queue

F and R are the front and rear in circular queue respectively

Flag is a variable for congestion notification

Set Queue length = Q

Set min = ⌊(2/3 of Q)⌋

    max = ⌈(4/5 of Q)⌉

While

         Packets are being transmitted in the network &

received at the routers.

do

        Set Flag=0;

       [Calculate no. of packets]

if (F<R)

      N= [(Q-F)+R+1]

else

    N=R-F+1

[Compare No. of packets with ' min' & ' max']

if (N ≥ min & N<max)

then

       [Packet arrival rate is greater than packet

       service rate]

[Congestion may occur]

     A packet will be send to sources to reduce

    transmission rate.

      Set Flag=1 [1 indicates Congestion]

elseif (N ≥ max)

then

     [Packet arrival rate is much greater than
```

packet service rate]

[Congestion will definitely occur in

sometime]

Router will drop 1 packet after every

specific interval of time.

else

[Packet arrival rate is smaller than packet

service rate]

No congestion

End if

End while

3.4 Hybrid-ERED (Effective RED) Fuzzy based Congestion Control

In these ERED is the one of the method of AQM (Active Queue Management) normally these methods are able to detect congestion in early stage and control it by packet dropping. This ERED method among many other AQM methods, gives a good performance in detect and control congestion and preserve packet loss.

In the ERED method with the FUZZY Inference process that eases the problem of parameter initialization and parameter dependency.

INITIALIZATION:

avg:= 0

count:= 1

FOR EACH arrival packet

CALCULATE new avg as follows:

IF q==0 THEN avg:=$(1\text{-w})^{f(\text{time-q\_time})}$ *

avg

IF q != 0 THEN avg:= (1-w)* avg + w *q

CALCULATE D and its related parameters as follows:

IF q:= FULL THEN $arr_p = arr_p\text{-}1$ and

$dep_p := (1\text{-}w_{dep}) * dep_{p\text{-}1} + w_{dep} * \#departed_p$

IF q!= FULL THEN

$arr_p := (1\text{-}w_{arr}) * arr_{p\text{-}1} + w_{arr} * \#arrived_p$ and

$dep_p := (1-w_{dep}) * dep_{p-1} + w_{dep} *$

$\#departed_p$

$D := (1/ dep_p) * arr_p * q$

CALCULATE Dp and its related parameters, and implements packet dropping, as:

if $(min_{th2} \le avg < max_{th3})$ && $(q \ge min_{th2})$

increase count

$Dp' = max_p * (avg - min_{th})/(max_{th} - min_{th})$

$Dp = Dp'/ (1-count * Dp') + w_d(D)$

with probability Dp

drop packet

count := 0

else if $(avg < min_{th2})$ && $(q > max_{th2})$

Calculate $Dp = max_p/(1-count * max_p)$

with probability Dp

drop packet

count := 0

else if $(avg \ge max_{th3})$

Drop packet

Count = 0

else

Count = -1

When q==0

q_time=time

3.5 Cross Layer Based Congestion Control

Cross Layer Congestion Control it is developed on of the basic concept of cross layer optimization. The main purpose of Cross Layer Congestion Control for different environments of traffic models. Such as single source to single destination and multiple source to multiple destinations.

Cross Layer congestion control algorithm based on joint power control in mac layer and congestion control in TCP layer for Reno-2 explained the following algorithm

Pseudo code for Cross Layer Congestion Control Algorithm Reno-2

Set initial transmission rate: $x_{i} = x_{intial}$

Initialize $Pl = Pl_{Min}$

Advertise the minimum $SINR_l$ required

Update G_{li} and G_{jj} periodically or after receiving the advertised signals

Determine maximum capacity

Determine $\lambda_l(t)$

Determine $m_j(t)$

Calculate $P_l(t+1)$

if $|P_l(t + 1) - P_l(t)| \leq \delta$

then

Continue transmission at $P_l(t)$

else

Transmit at min $(P_l(t + 1), P_{lMax})$

end if

Change according to the congestion control algorithm

of Reno-2

Update $SINR_l$ at each node go to Step 3

3.6 TCP using RED with SNR ratio

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 window before congestion happen.

In digital communication a Signal-to-noise (SNR) is a measure used to compare the level of a desired signal to the level of background noise higher numbers.

In this RED algorithm when operating in wireless mode exploits the SNR ratio of the communication lines to decide whether a timed out packet was due to congestion or error loss. When TCP connection first begins, the alternative RED algorithm computes the average of the queue. Then it checks the average is greater than the threshold algorithm checks the reserved bits which indicates the connection is wired or wireless.

Pseudo code for Algorithm

mark the arriving packet

count ← 0

else b=1 then wired

if SNR>5dB

mark the arriving packet

count ← 0

else SNR<5dB

mark the arriving packet

count ← 0

retransmit the packet

else count ← -1

when queue becomes empty

q_time ← time

Saved variables

avg: average queue size

q_time: start of the queue idle time

count: packets since last marked packet

Fixed parameters

W_q: queue weight

Min_{th}: minimum threshold for queue

$maxt_h$: maximum threshold for queue

max_p: maximum value for pb

Others

pa : current packet-marking probability

q: current queue size

time: current time

4. Advantages and Disadvantages of Congestion Control Algorithms

S.no	Algorithms	Advantages	Disadvantages
1	URED	It gives higher throughput and lower packet drops in normal flow and congestion due to DDoS flood attack flow	Most of sites do not expose that they were attacked to avoid denigration
2	Multipath Congestion Control	The cause of packet loss helps MP Veno to avoid unreasonable half reduction	The performance of MP Veno depend on choice of threshold

		of congestion window over wireless link	
3	Hop-by-Hop	This algorithm helps predicting congestion and improving efficiency and reliability	In this algorithm executes control mechanisms based on the feedback
4	Hybrid ERED	The performance of the fuzzy-based form give better throughput than the parametric-based form and ERED in terms of delay and packet loss	In the hybrid-ERED, a few scenarios are consider that are based on the status of the indicators according to some threshold
5	Cross Layer Congestion Control	The cross layer congestion control technique provides stabilized throughput at low power transmission	The channel conditions are very bad, which in turn result in an increase in power transmission
6	RED with SNR	In this algorithm the SNR ratio uses to detect the reliability of the link proactively	This RED Algorithm drops the packet when the congestion is occurring

5. Conclusion

Congestion is the one of the very important issues in related to network. This paper briefly gives various types of congestion control algorithms. Every congestion control algorithm solves one unique problem. It seems that at present there is no single algorithm that can resolve all of the problems of congestion control on computer networks.

6. References

1. Chandini M Patel, URED: Upper Threshold an Efficient Congestion Control Algorithm, 4th International Conference on Computing Communication and Networking Technologies, Page no:1-5, Year-2013
2. Tuan-Anh Le, Improving the Performance of Multipath Congestion Control over Wireless Networks, The 2013 International Conference on Advanced Technologies for Communications, page no: 60-65, Year-2013
3. Urvashi Handa and Padma Bonde, Improving Hop-by-Hop Congestion Control Algorithm, International Journal of Computer Science and Information Technologies, Volume no:6, Page no: 5505-5508, year-2015
4. Maimuna Khatari and Ghassan Samara, Congestion Control Approach based on Effective Random Early Detection and Fuzzy Logic, Volume no:8, Page no: 180-193, year-2015
5. Hemant Kumar Rath and Anirudha Sahoo, Croos Layer Based Congestion Control in wireless Network
6. Shanmuga Priya and Dr.N.R.Ananthanarayanan, A Study on TCP Congestion Control Using RED Algorithm with SNR Ratio, International Journal of Computer Science, Volume no: 2, page no: 028-034, year-2014