International Journal of Advance Research in Science and Engineering
Vol. No.6, Issue No. 08, August 2017
www.ijarse.com
IJARSE
ISSN (O) 2319 - 8354
ISSN (P) 2319 - 8346

End to End Congestion Control Policies Behavior Evaluation: An Experimental Study

Deepak Mehta[1], Dr. Yogesh Kumar Sharma[2], Dr. Chaman Verma[3]

Research Scholar, Deptt. of Computer Science, JJT University, Rajasthan (India)

Associate Prof., Deptt of CSE, JJT University, Rajasthan (India)

Assistant Professor, Deptt. of Computer Science, JCD Vidyapeeth, Sirsa, Haryana (India)

ABSTRACT

Due to compatibility of TCP with different congestion control techniques inspire the researcher to review various alternative congestion control algorithms. One of them congestion control algorithm is slowly responsive which can be characterized by slow response as against the behavior of tradition TCP congestion control like Reno, New Reno etc. on single packet loss. Steady state in TCP compatible congestion control algorithm described as static state on fixed loss. This article is going to investigate the dynamic behavior of different slow responsive TCP compatible congestion control algorithms on the basis of performance metrics.

Keywords: AIMD, ECN, Slow-Response, Congestion, Steady-State

I. INTRODUCTION

Router role in any congestion control scenario is passive and they are literally only for indicating congestion by losing packet or by coating mark of ECN. Routers do not take responsibility of bandwidth control but this is done by end host. Overall responsibility is only of host that respond accordingly to these congestion signals, thus active host and passive router collectively in tandem manner playing vital role in congestion management asdiscussed by Jacobson(1988).The article discuss here the existenceof congestion control mechanisms which customized for versatile applications requirement in TCP controlled world like TCP-Friendly Rate Control (TFRC), equation based AIMD congestion control with TCP linear constants and binomial congestion control. These congestion control mechanisms protocols characterized as slowly responsive and they do not halves their transmission rate as like TCP congestion control like Reno, SACK on single packet loss alsodescribed by Floydd S., Handley (2000) and Bansal and Hari Balkrishan (2001).In this article our objective is to make the behavior evaluation of various slowly responsive TCP-Compatible Congestion control techniques on the basis of static and dynamic condition by using various performance metrics like loss rate, fairness properties of long and short –term, utilization of connection.

II. EXISTING CONGESTION CONTROL MECHANISMS

Mostly TCP connection in steady-state environment adopt two congestion control mechanism named Additive Increase Multiplicative Decrease (AIMD) which dictate the size of transmission window and other is self-

IJARSE
ISSN (O) 2319 - 8354
ISSN (P) 2319 - 8346
clocking which work on the packet conservation principle by taking control of transmission window size and data transmission.

End-to-End congestion control mechanism based on steady-state environment can be categorized as TCP-Compatible, TCP-equivalent and TCP-Incompatible congestion control. Other classification can be made on the basis of transient state behavior are slowly-responsive congestion control protocolsand fast TCP congestion control. This is also illustrates in below figures1 and 2.

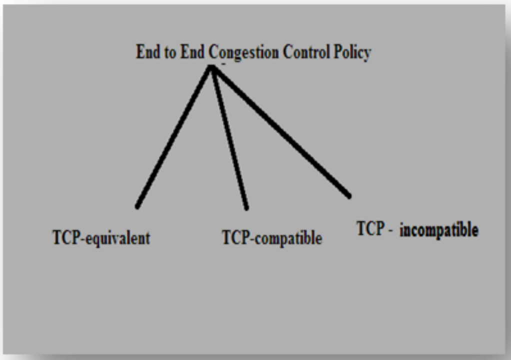


Fig 1. E2 E Congestion Control based on steady-state

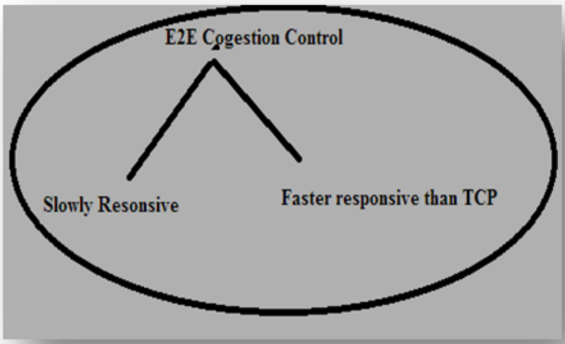


Fig 2 E2E Classification on transient response

In TCP-Equivalent policy, it usesadditive increase and multiplicative decrease (AIMD) method to dictate transmission with increase and decrease aswork like in TCP variants case as discussed by Yang Richard (2000), also Rate Adaption Protocol (RAP) which uses rate base equation without self-clocking scheme.TCP-Compatible congestion control protocols are characterized by same throughput generation on different round-trip time (RTT) in steady-state environment.

Slow Responsive congestion control mechanism can be defined in context of TCP as decreased transmission rate on single-packet loss.As name implies reaction to congestion is small means less sensitive as compare to TCP AIMD approach. If TCP-slow responsive mechanism compared to the faster TCP responsive policy, response to individual packet drop is very slow and this will allow application to give benefit of smoother sending rate. TFRC, AIMD with different increase and decrease TCP constants are slow-responsive congestion control policies. This can be understood with below described algorithm of AIMDin figure 4 which have two parameters for dictating the size of window (X,Y). It can be TCP Compatible and slow-responsive depend on the Y parameter value, if Y value is less than or equal to 0.5then it is both Slow-Responsive and TCP Compatible else then it will be only TCP-Compatible.There isvariable called CWND_SZ for transmission window and X value is 1 and Y value is considered as 0.5 provided without delayed acknowledgment in TCP. If want to be TCP-compatible then X and Y value not treated independent and X value depend on the Y value by following equation with Y value less than 0.5.This can be named with slowly responsive AIMD(Y) algorithm.

$$X = 4 * (2 * Y - Y^2)/3$$

IJARSE
ISSN (O) 2319 - 8354
ISSN (P) 2319 - 8346

Other class of congestion control algorithm is non-linear congestion control called binomial algorithms. It is mostly found that application of streaming audio and video, transmission rate is reduced at extreme rate on the occurrence of congestion in TCP which are problematic for users.Binomial algorithms generalize TCP AIMD style policy. Increment is made on the inversely proportional to a power of I of the current window and reduction is made also of power of J for TCP is 1 as discussed and described by Deepak Bansal and Hari Balkrishan (2001). As it is binomial congestion control policies if I+J value not equal to 1 then it is in slow responsive but if both parameter value addition equal to 1 then it comes under slow responsive TCP compatible congestion control policies. RAP and TCP are both come under TCP-equivalent policies provided steady-state environment is present.

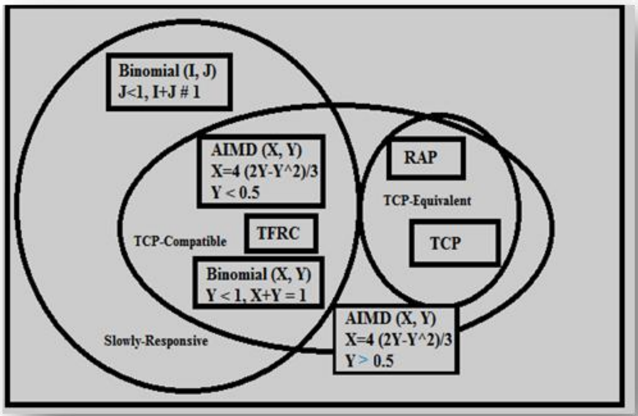


Fig3 Relationship among E2E congestion policies

AIMD (X ,Y)

1. If loss encountered Then
   a. CWND is decreased CWND_SZ to (1-Y)*CWND_SZ
   b. Return

[End of If Structure]

2. CWND_SZ=CWND_SZ to CWND_SZ+X [in the absence of packet-loss]
3. Return

Fig 4 AIMD Algorithm working

Binomial (A, B, C, D)

1. If congestion encountered Then
   a. CWND_SZ = CWND_SZ – D*CWND_SZ^{B}
   b. Return

[End of if Structure]

2. CWND_SZ= CWND_SZ+ C/CWND_SZ^{A} [if congestion not occur]
3. Return

Fig 5 Binomial Algorithm Working

Binomial congestion control algorithm is basically depend on four parameters named as by hypothetical manner A,B,C,D and have property of non-linear generalization of AIMD algorithms. Its algorithm is illustrated in fig 5. It is TCP-Compatible only when A+B =1 and B value is less than or equal to 1 and will be slowly responsive only when B is less than 1.

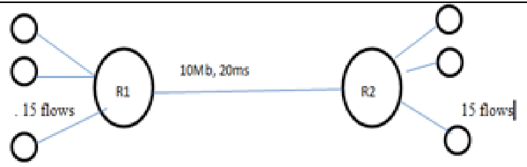
IIAD (A=1 and B=0) and SQRT (A=1 and B=0.5) are famous studies Slowly-responsive TCP-Compatible congestion control algorithms as smaller value of B force the algorithm to be slowly responsive than larger

IJARSE
ISSN (O) 2319 - 8354
ISSN (P) 2319 - 8346

value.Other method to deal with the loss event is respond to loss after some interval of time using TFRC CC protocol. Floyd (2000)suggested making TFRC to TCP-Compatible by using the TCP's sending rate function which generates response of event rate of loss and RTT. TFRC(x) function estimates the loss event at average rate with loss intervals of x over latest period of time.

III. EXPERIMENT EVALUATION OF SLOW CC ALGORITHMS

In experiment evaluation the present work going to investigate the performance various slow congestion control (Slow CC) algorithms with NS2 simulator in the presence of sudden bandwidth reduction by introducing high volume of traffic generated by sudden crowd of many flows of TCP and also CBR traffic generated at said below intervals . Therefore it stimulates sudden increase in congestion impulse packet loss. Table 1 Slow CC Protocols Simulation Environment Parameters

Simulator	NS2
Stop Time	250 second
Traffic Generation	15 long lived Slow CC Flows with random start timing (FTP traffic generation used) 1 CBR traffic generator
Topology	
CBR Source ON/OFF timing	T=0 to 100 (ON state) After some interval Restart at T=150
Flow Used	TFRC without self-clocking TFRC with self-clocking TCP AIMD(1/2) TCP AIMD(1/256) Binomial type SQRT(1/256)
Active Queue Management	Random Early Detection (RED) at bottleneck bandwidth
Performance Metric	Queue Average Packet Loss Rate

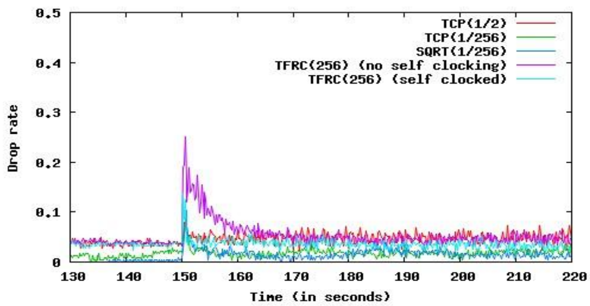


Fig 6 Loss Rate for various Slow CC algorithms

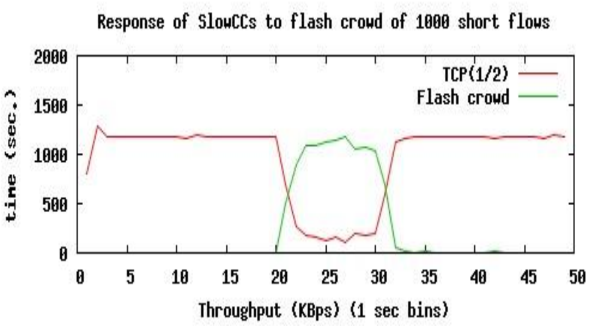


Fig 7 Slow CC Responses Towards Flash Crowd 100 Short Flow

IJARSE
ISSN (O) 2319 - 8354
ISSN (P) 2319 - 8346

RED active queue management technique is used on bottleneck link. On the beginning of traffic generation of all sources using 15 long lived flows and CBR traffic generators, all sources utilized the bandwidth. During idle period of time between 100 to 130 drop rate is negligible but when CBR traffic generated next time i.e. on 150 the drop rate is increased of flows and end to end congestion control algorithms take advantages of this control. As figure 6 and 7 illustrates all slow congestion control algorithms uses very slow response even packet loss generation start at 130 but response to loss event taken them at 150.The present work also experimented with more real scenarios where intense bandwidth decrease is marked by sudden small HTTP traffic crowd generations also call web transfer traffics with the new source of CBR. The web transfer traffic initialized at the time 20 with short transfer streams of TCP (10 data packets) arriving at 100 flow/sec for 10 sec. All throughputs estimating figures 8 to figure 14 illustrates the ideas of all slow CC algorithms. All below figures exhibits aggregate throughput got by small and long live slow congestion control protocols. TCP (1/2), TCP (1/8), TCP (1/256), TFRC (8), TFRC (256) without self-clocking, TFRC (256) with self-clocking, IIAD (8) and SQRT (1/2) are the slow-congestion control examples with flash crowd traffic generated at 20.

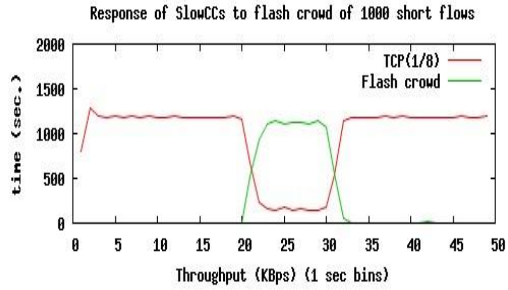


Fig 8 Slow CC Responses to Flash Crowd 1000 Short flows with TCP (1/8)

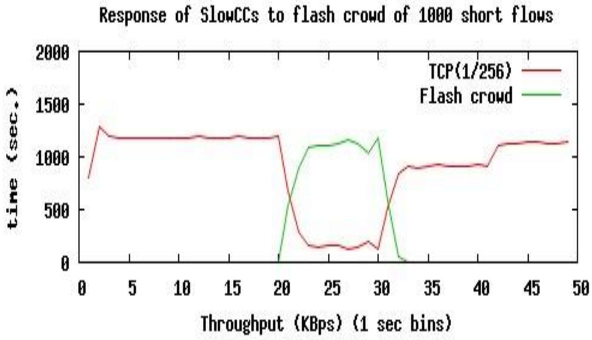


Fig 9 Response Depiction Slow CC to Flash Crowd of 1000 Short Flows with TCP(1/256)

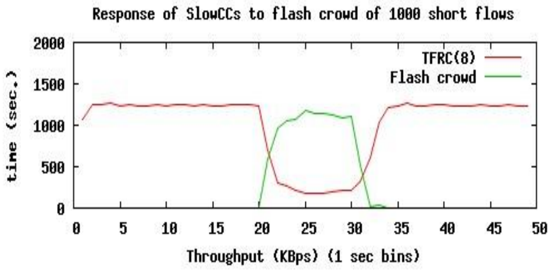


Fig 10 Response Depiction of Slow CC to Flash Crowds of 1000 short flows with TFRC(8)

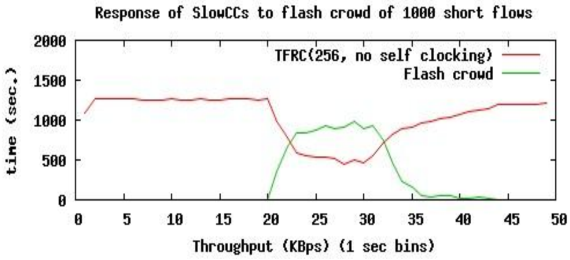


Fig 11 Response of Slow CC to Flash Crowds of 1000 short flows with TFRC(256) and without self-clocking

International Journal of Advance Research in Science and Engineering
Vol. No.6, Issue No. 08, August 2017
www.ijarse.com
IJARSE
ISSN (O) 2319 - 8354
ISSN (P) 2319 - 8346

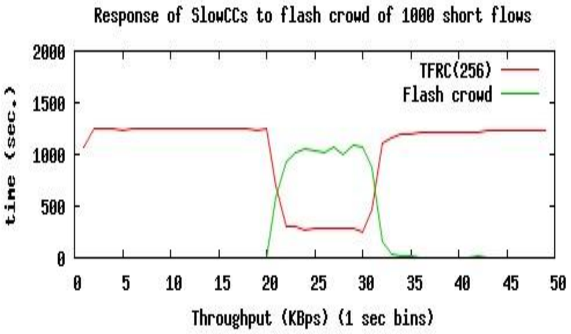


Figure 12 Response of Slow CC to Flash Crowds of 1000 short Flows with TFRC(256)

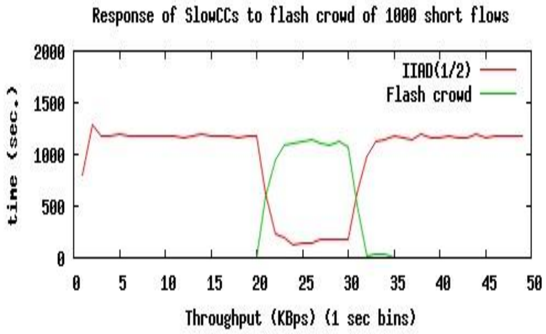


Figure 13 Response of Slow CC to Flash Crowds of 1000 short flows with IIAD (1/2)

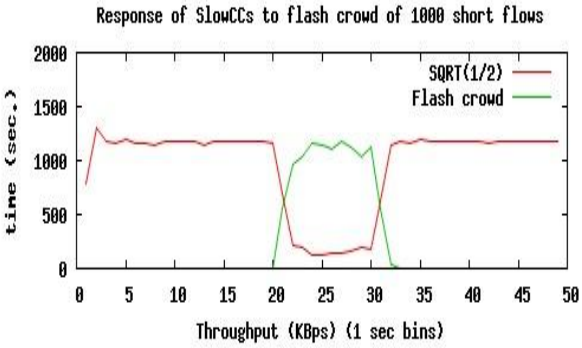


Figure 14 Slow CC Aggregate Throughput for long lived flows with SQRT (1/2)

From the above figures TFRC with self-clocking, SQRT and IIAD respond earlier to respond to flash FTP traffics.

IV. CHALLENGES OF SLOW RESPONSE CONGESTION CONTROL ALGORITHMS

TCP-compatible slow CC algorithm not deployed in real world traffic due to it unfairness to other congestion control protocols like TCP and do not support 100% utilization of bottleneck bandwidth. This can be proved by long and short term fairness performance metrics in dynamic environment.

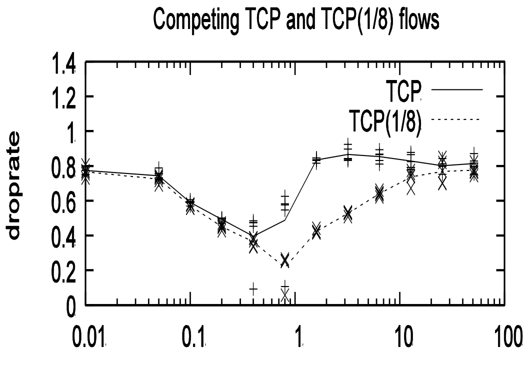


Fig 15 TCP and TCP (1/8) w.r.t simulation time

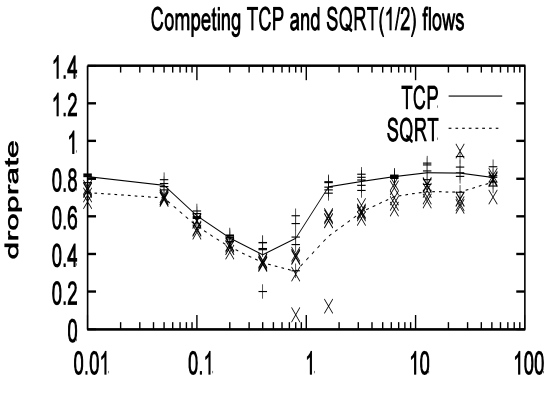


Fig 16 TCP and SQRT (1/2) flows w.r.t Simulation Time

IJARSE
ISSN (O) 2319 - 8354
ISSN (P) 2319 - 8346

To verify the fairness of slow-congestion control algorithms in environment which is dynamic in nature increases bandwidth periodically thrice of available bandwidth. Other parameters also have changed to challenge the environment of execution as follows.

Table 7.2 Slow CC TCP and TCFRC Protocol With flows Simulation Parameters

Flow type	TCP, TFRC
No. of Flows	Total 14 flows 7 TCP 7 TFRC
Allotment of Bandwidth	Total 15 Mbps 5 Mbps allotted to long-lived flow when CBR source is active

Here our purposed work makes use of an ON/OFF source of CBR with equal ON-OFF times exhibiting the repeating Square Wave pattern from available of bandwidth used. These scenarios (Stress Tests) are not for accurately model as real scenario but it is used to explore the benchmark of slow congestion control mechanism in given environment.

In the figures 17 to figure 19 we are going to monitor the related fairness of TFRC and TCP as a period function of source CBR. In figure 17, here column represent the result from simulation with single mark illustrating ten flow throughputs by marking drop rate, if drop rate is low it will illustrate high throughput. X-axis displaying combined high and low bandwidth period with second units' length and throughput exhibit by y-axis of available bandwidth which is normalized by fair share of single flow. Both lines display the usual throughput got by TFRC and TCP flows. In below figures., when the CBR source period is at 0.2 second or can be said as when RTT is 4 then total efficient use of bottleneck link is become high . When source of CBR lie between 1 and 10 sec the TCP flow receives high throughput and TFRC flows displayed dynamic environment in which conditions of network that favors to TCP but not to TFRC. To find the situations or conditions in which TFRC may compete for TCP unfairness and CBR sources include saw tooth pattern from where the sources of CBR entered into and also in off period becoming less slow it transfer rate to an off period.

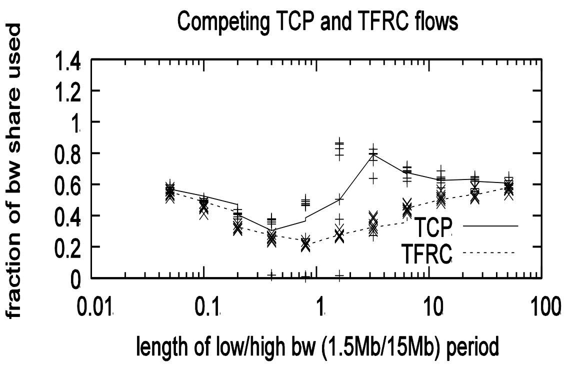


Fig 17 TCP and TFRC flows w.r.t Simulation Time over the length low/high Bandwidth

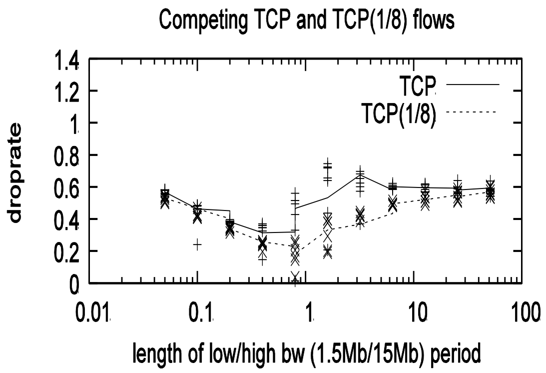


Fig 18 TFRC and TCP (1/8) w.r.t Simulation Time with Low/High Bandwidth

IJARSE
ISSN (O) 2319 - 8354
ISSN (P) 2319 - 8346

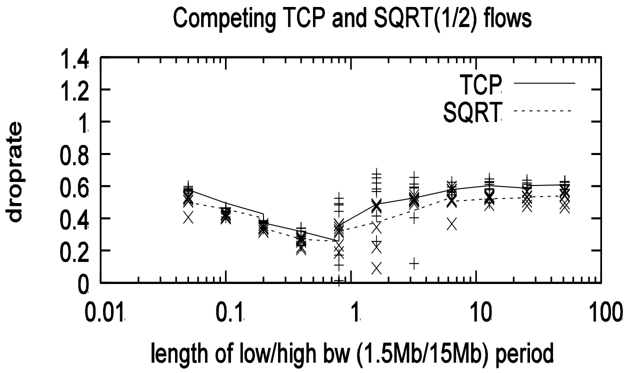


Fig19 Low/High BW TCP and SQRT (1/2) flows w.r.t time

Therefore slow congestion control techniques includes the TFRC, AIMD with supportive and varied constants, binomial techniques with experimented performance metrics evaluation promise for safe deployment with some exceptional problems.

REFERENCES

1. Jacobson, Van.(1988). Congestion avoidance and control. In *ACM SIGCOMM computer communication review*, vol. 18, no. 4, pp. 314-329. ACM, 1988.
2. Bansal, Deepak, and Hari Balakrishnan.(2001). Binomial congestion control algorithms. In Proc. Of INFOCOM 2001. Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE, vol. 2, pp. 631-640. IEEE, 2001.
3. Floyd, S. (1994). TCP and explicit congestion notification. ACM SIGCOMM Computer Communication Review, 24(5), 8-23.
4. FLOYD, S., HANDLEY, M., PADHYE, J., AND WIDMER, J.(2000). Equation-Based Congestion Control for Unicast Applications. In Proc. Of SIGCOMM Symposium on Communications Architectures and Protocols (Stockholm, Sweden, August 2000), pp. 43–56.
5. Floyd, Sally, and Kevin Fall.(1999). Promoting the use of end-to-end congestion control in the Internet. IEEE/ACM Transactions on Networking (ToN) 7.4 (1999): 458-472.
6. Floyd, Sally, Mark Handley, and Jitendra Padhye.(2000). A comparison of equation-based and AIMD congestion control. (2000).
7. Handley, M., Floyd, S., Padhye, J., & Widmer, J. (2002). TCP friendly rate control (TFRC): Protocol specification (No. RFC 3448).
8. Hassan, Suhaidi, and Mourad Kara.(2000). Simulation-based performance comparison of TCP-friendly congestion control protocols. In Proc. of the 16th Annual UK Performance Engineering Workshop (UKPEW2000). 2000.
9. Yang Richard, and Simon S. Lam. (2000). General AIMD congestion control. In Proc. 2000 International Conference on Network protcol, pp. 187-198.