Comparative study of TCP friendly Window based Congestion Control Algorithms

Kulvinder Singh

Asst. Prof., Computer Science and Engineering Department,
Vaish College of Engineering,
Rohtak, India
kulvinderbhuna@gmail.com

Abstract

This work presents a thorough study of TCP like window based congestion control. These schemes are basically different from memory less AIMD and binomial schemes. As like this, a small step allows a much broader exploration of friendliness for TCP congestion controls than memory less AIMD and binomial schemes. We propose An Array of window based congestion controls possessing higher smoothness in steady states, while it reacts on promptly to sudden changes in network conditions. We analyze the smoothness, transient behavior and performance comparisons for new Array of control. In SIMD (Square Increase Multiplicative Decrease), congestion window size increases superlinearly in proportion to the square of time elapsed since the detection of last loss event. In other words, increase is inversely proportional to window size at the time of last loss detection. Thus, SIMD has higher aggressiveness and fast convergence to fairness.

Keywords: Congestion, TCP, aggressiveness, friendliness.

1 Introduction

TCP uses Additive Increase and Multiplicative Decrease (AIMD). It probes available bandwidth by increasing congestion window size and respond to increase congestion by packet losses. In other words, by decreasing the window size multiplicatively. Recently proposed congestion control mechanisms, includes the generalizations of TCP like window based schemes[1, 2, 3, 4], and equation based schemes[5, 6, 7]. The common objectives of these schemes is to reduce high variability of TCP transmission rates. Such high variability may limits the network utilization. It is not desirable for emerging applications such as real time streaming applications on the Internet.

A new transport protocol should be implemented congestion control mechanisms that interact well with TCP [8]. It should maintain TCP compatibility or fairness across the connections using different types of protocols. To provide such fairness, one solution is to satisfy the TCP friendliness. Which means the (λ, p) relationship $\lambda \sim 1/(R\sqrt{P})$ should hold. Where λ is throughput of a flow, p is loss rates and R is the round trip time (RTT).

In addition, TCP friendliness, smoothness, aggressiveness and responsiveness [1, 9] are the important indices of performance in congestion control. Smoothness indicates variability of transmission rate. Aggressiveness indicates how much fast a connection probes extra bandwidth by opening up its congestion window. Responsiveness measures how fast a connection reacts for increase congestion by decrement of its window size. Smoothness characterizes the steady state behavior of congestion control protocols. Both aggressiveness and responsiveness characterize transient behavior. The important observation is that there are comparisons among smoothness, aggressiveness, and responsiveness [1, 9]. Comparisons of TCP, general AIMD [1, 3], TFRC have shown that higher smoothness means less aggressiveness and responsiveness.

2 TCP Friendliness

We expose that our management body structure using the management rules can be TCP friendly. The aspect of TCP friendliness comprises the partnership between throughput and supply reducing pace. We believe a unique step down design, where the cuts are Bernoulli tests; packages are decreased systematically with a resolved possibility.

Assuming such a unique reduction design, and without conceiving the result of TCP's timeout systems, we depict the use of the following differentiation of α to create our congestion control schemes TCP friendly:

(k, l)	Increase rule	Decrease rule	Increase function
$k = 0, l = 1$, AIMD	$w_{t+1} \leftarrow w_t + \frac{3\beta}{2-\beta}$	$w_t \leftarrow w_t - \beta w_t$	$w(t) = w_0 + \frac{3\beta}{2-\beta}t$
$k = -\frac{1}{2}, l = 1$, SIMD	$w_{t+1} \leftarrow w_t + \frac{3\sqrt{\beta}}{\sqrt{2(1-2\beta/3)}}\sqrt{\frac{w_t - w_0}{w_{\max}}}$	$w_t \leftarrow w_t - \beta w_t$	$w(t) = w_0 + \frac{9\beta}{8(1-2\beta/3)^2}\frac{1}{w_{\max}}t^2$
$k = 0, l = \frac{1}{2}$	$w_{t+1} \leftarrow w_t + \frac{3\beta}{2\sqrt{w_{\max}}-\beta}$	$w_t \leftarrow w_t - \beta\sqrt{w_t}$	$w(t) = w_0 + \frac{3\beta}{2\sqrt{w_{\max}}-\beta}t$
$k = 0, l = 0$, AIAD	$w_{t+1} \leftarrow w_t + \frac{3\beta}{2w_{\max}-\beta}$	$w_t \leftarrow w_t - \beta$	$w(t) = w_0 + \frac{3\beta}{2w_{\max}-\beta}t$

Table 1: Several special cases of TCP friendly congestion controls.

$$\alpha = \frac{3}{2(k+1)\left(1-\frac{1}{k+2}\beta w_{max}^{l-1}\right)}\left(\frac{\beta}{\Gamma(\frac{1}{k+1}+1)}\right)k + 1w_{max}^{kl+l-1} \quad (1)$$

where the Gamma function Γ(.) is a constant. We accept:

$$c = \left(\frac{3}{2\left(1-\frac{1}{k+2}\beta w_{max}^{l-1}\right)}\right)^{\frac{1}{k+1}}\left(\frac{\beta}{\Gamma(\frac{1}{k+1}+1)}\right)w_{max}^{l-\frac{1}{k+1}} \quad (2)$$

When the window size variation is small i.e. window decrease is small, $\beta\, w_{max} << w_{max}$, we can simplify α and c as:

$$\alpha \approx \frac{3}{2(k+1)}\left(\frac{\beta}{\Gamma(\frac{1}{k+1}+1)}\right)^{k+1}w_{max}^{kl+l-1} \quad (3)$$

$$c \approx \left(\frac{3}{2}\right)^{\frac{1}{k+1}}\left(\frac{\beta}{\Gamma(\frac{1}{k+1}+1)}\right)^{w_{max}^{l-\frac{1}{k+1}}} \quad (4)$$

That is, α is a constant factor of w_{max}^{kl+l-1}, and c is a constant factor of $w_{max}^{l-\frac{1}{k+1}}$.

Table 1 shows several special cases. We show their control rules and window increase functions. When k = 0 and l = 1, from (1) we have $\alpha AIMD = 3\beta/(2 - \beta)$. If $\beta<<1$, $\alpha AIMD \approx 3\beta/2$. It degenerates to the memoryless TCP friendly AIMD control [1, 3]. When k = −0.5 and l = 1,

$$\alpha_{SIMD} = \frac{3\sqrt{\beta}}{(1-\frac{2\beta}{3})\sqrt{2w_{max}}} \quad (5)$$

If $\beta<< 1$, $\alpha_{SIMD} \approx \frac{3\sqrt{\beta}}{\sqrt{2w_{max}}}$.

In that case, the window sizing decrements multiplicatively upon the detecting by packet loss, but increments proportionate the square of the clock time elapsed since the detection of the last loss effect (Table 1). We address this control SIMD (Square Increase/Multiplicative Decrease).

Additional way of instancing TCP friendliness is to compare our controls with binomial controls. In [4], the authors showing that binomial controls are TCP friendly. We detect that for all instance of the binomial controls, there is a comparable point along the line where k = 0 and 0 ≤ l ≤ 1. Which approximately gives the same control rules. For example, the point k = l = 0 corresponds to the particular case IIAD (Inverse Increase/Additive Decrease) of binomial controls. IIAD accepts the following control rules:

Increase : $w_{t+1} \leftarrow w_t + \frac{3\beta}{2w_t}$

Decrease : $w_t \leftarrow w_t - \beta$.

The only deviation between IIAD and our AIAD is in the window increment factor: in IIAD, the factor is inversely proportional to the current window size wt, although in AIAD, the factor is a constant whose measure is inversely proportional to w_{max}. Notice that w_{max} shows the maximum window size in the former congestion era, so its value is relative to the time average of wt if the TCP congestion window has accomplished steady state. In additional words, IIAD and AIAD controls are like in steady state. However, when there is an abrupt increase in network bandwidth, AIAD's additive increase rule comprises more aggressive than the IIAD's sub linear increase rule.

The preceding observance applies to all cases of binomial controls, with only one exclusion at k = 0, l = 1, i.e. AIMD control, where our control algorithm deviates exactly to general AIMD.

	Smoothness	*1/Aggressiveness*	*1/Responsiveness*
AIMD	$\frac{0.41\beta}{1-\beta/2}$	$\frac{m-1}{\beta}\frac{2W}{3}$	$\log_{(1-\beta)}\frac{1}{m}$
IIAD	$\frac{0.41\beta}{W-\beta/2}$	$\frac{(m^2-1)W^2}{3\beta}$	$\frac{W(1-1/m)}{\beta}$
SIMD	$\frac{0.73\beta}{(1-2\beta/3)^2}$	$\sqrt{\frac{m-1}{\beta}\frac{2\sqrt{2}W}{3}}$	$\log_{(1-\beta)}\frac{1}{m}$
AIAD	$\frac{0.41\beta}{W-\beta/2}$	$\frac{2(m-1)W^2}{3\beta}$	$\frac{W(1-1/m)}{\beta}$

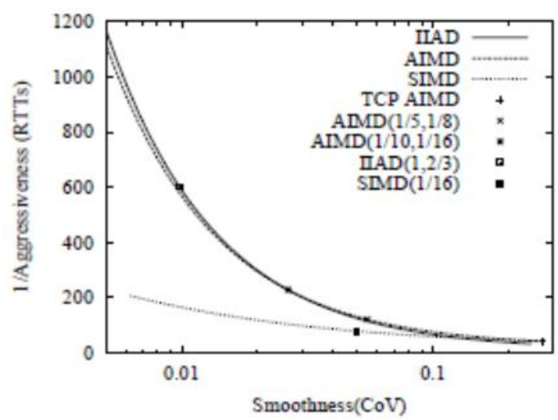
Table 2: Smoothness, Aggressiveness and Responsiveness comparisons of AIMD, IIAD, SIMD and AIAD.

3 Comparisons between Smoothness, Aggressiveness, and Responsiveness

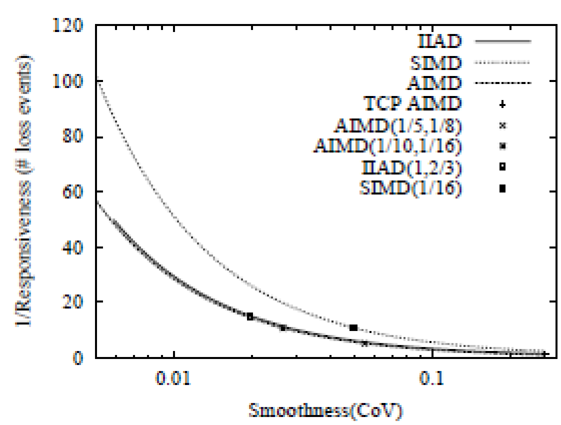
In this segment, we consider significant properties of congestion controls other than TCP friendliness. These are smoothness, aggressiveness, and responsiveness. Smoothness assesses the variableness in a connection's window size across time. Higher variance is not desirable. Aggressiveness assesses how fast a connection investigations bandwidth as it gets available by opening up its window. Broader aggressiveness, involving potentially higher utilization, is desirable. Responsiveness assesses how fast a connection drop offs its window size in response to increased congestion. High responsiveness is worthy.

Smoothness can be followed at different time plates [1]. We consider short time plates since long term smoothness can be involved by other dynamics in the system. We define smoothness as the variant of the window size of a joining during one congestion era. In detail, we use the coefficient of variation of window size in one congestion epoch as an amount of short term smoothness. Note that the coefficient of variance is not of necessity an exact measure of smoothness, but it is equal to give insight into the comparisons. We specify aggressiveness as the inverse of the clock time needed for the connection to increase the window size, in reaction to a step increase of available bandwidth [9]. That is, the available bandwidth is expanded by a factor of m. We define responsiveness as the inverse of the number of expiration events required for the connection to step down its window by a significant amount, in response to a step increase of congestion [9]. That is, a decrease of available bandwidth by a factor of m.

Table 2 gives the estimate expressions of smoothness, aggressiveness, and responsiveness for AIMD, IIAD, SIMD, and AIAD controls. More details are given in [13]. Intuitively, the smoothness index is proportional to the window drop off divided by the average window size. Aggressiveness is ascertained by the window size increase function. Responsiveness is decided by the decrease rule.

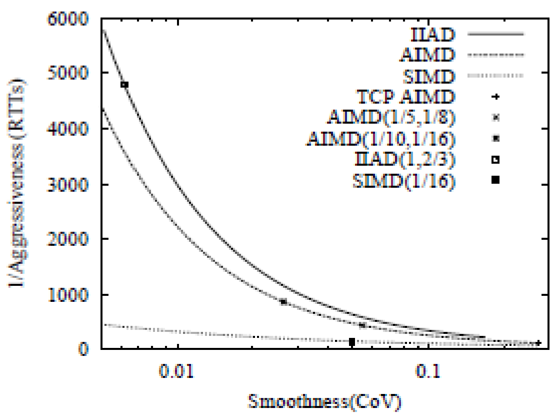


(a) Aggressiveness vs Smoothness.

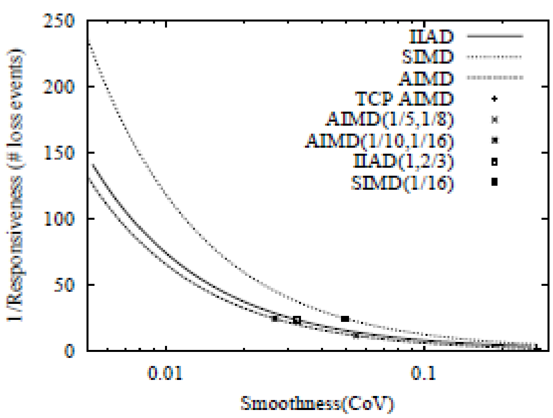


(b) Responsiveness vs Smoothness.

Figure 1: Comparisons of smoothness, aggressiveness and responsiveness. For (a), we assume available bandwidth is double. For (b), we assume window is reduce to half, i.e., m = 2. The initial average window size, W, before bandwidth changes to 20.



(a) Aggressiveness vs Smoothness.



(b) Responsiveness vs Smoothness.

Figure 2: Comparisons of smoothness, aggressiveness, and responsiveness. The configurations are same as those of Figure 1 except that the bandwidth decrease or increase factor m = 5.

Mathematical results in Figure 1 show the comparisons among smoothness, aggressiveness, and responsiveness. Results for AIAD are not displayed here since they are alike to those of IIAD exclude that AIAD has higher aggressiveness. Figure 1(a) displays the inverse of aggressiveness of AIMD, SIMD, and IIAD as the coefficient of variation varies. Their particular cases TCP, AIMD(1/5,1/8), AIMD(1/10,1/16), IIAD(1,2/3), and SIMD(1/16) are too shown by points. Note, AIMD(1/5,1/8) and AIMD(1/10,1/16) are parameterized according to the TCP friendly condition $\alpha = 3\beta/(2 - \beta)$. The inverse of aggressiveness is calculated as the number of RTTs essential to double the window size, i.e., m = 2. Figure 1(b) shows the inverse of responsiveness of AIMD, IIAD, and SIMD as the coefficient of variance varies. The opposite of responsiveness is computed adopting the target window size is half of the current window size, i.e., m = 2. From these figure, we could see that SIMD has much higher aggressiveness than the others, especially

While high smoothness (low coefficient of variance) is needed. In the meantime, SIMD has a slight deprivation of responsiveness. Particularly, SIMD demonstrates up to order of magnitude better aggressiveness at lower than about 1.7 times lower responsiveness for nearly the same smoothness value. For example, we can forecast that AIMD(1/20,1/30), SIMD(1/30), and IIAD(1,2/3) have comparable to smoothness when the mean window size is 20. However, SIMD(1/30) can respond to a substantial increase of available bandwidth much faster. The smoothness aggressiveness relationship could also be generalised from Table 2. For both AIMD and IIAD, aggressiveness changes corresponding the coefficient of variation. For SIMD, aggressiveness varies as the square root of the coefficient of variation. Thus, when the transmission rates are very smooth, SIMD has much highest aggressiveness than AIMD and IIAD.

We should mention that, we have not conceived the effect of the self clocking attribute of window based schemes in our analysis of responsiveness. When there is a burst of packet losses, since the connections are ACK clocked, it is imaginable that the congestion window size is came down to one due to a retransmission timeout regardless of which control is used. Hence, SIMD's slight loss of responsiveness is even little noticeable in such scenarios.

Figure 2 shows the same comparisons, exclude that we use a larger factor m = 5 for the sudden increment and decrement of available bandwidth. It appearances that the advantage of SIMD's aggressiveness is more articulated. We can also observe from Table 2 that, for SIMD, aggressiveness is inversely proportional to the square root of m, and for AIMD and IIAD, aggressiveness is inversely proportional to m or even m2 respectively. Therefore, larger m makes SIMD more favorable.

4 Convergences to Fairness and Efficiency

In this section, we firstly show the convergence of our SIMD example. Then we show that SIMD has best convergence behaviour than AIMD.

We acquire the synchronised feedback assumption [14]. This presumption is not naturalistic in real networks, and our analysis is not a validation of convergence if this assumption doesn't hold. However the analysis still allows for an intuitive fashion to gain insights. To show that multiple users with synchronised feedbacks using our control scheme converge to fairness, we apply the vector space used by Chiu and Jain [14] to view the system state modulations as a trajectory. For ease of demonstration, we show a two user case. It is straightforward to apply the same method to the multiple user case to reach the same conclusion.

Algorithm	T_1 (RTT)	Δ	T_2 (RTT)
TCP	$\frac{W-W_1-W_2}{2}$	$W_2 - W_1$	$\frac{W}{4}\log_{1/2}\frac{\epsilon}{\Delta}$
AIMD	$\frac{(W-W_1-W_2)(2-\beta)}{6\beta}$	$W_2 - W_1$	$\frac{(2-\beta)W}{6}\log_{1-\beta}\frac{\epsilon}{\Delta}$
IIAD	$\frac{1}{12\beta}((\frac{(W_2^2-W_1^2)}{W})^2 - 2(W_1^2+W_2^2)+W^2)$	$\frac{W_2^2-W_1^2}{W}$	$\frac{W}{3}\log_{1-2\beta/W}\frac{\epsilon}{\Delta}$
SIMD	$\frac{2}{3}(1-\frac{2\beta}{3})\sqrt{\frac{2}{\beta(1-\beta)}}\sqrt{\frac{W_1W_2(W-W_1-W_2)}{W_1+W_2}}$	$(2-\frac{W}{W_1+W_2})(W_2-W_1)$	$\frac{\sqrt{2}W}{3}\log_{1-2\beta}\frac{\epsilon}{\Delta}$

Table 3: Performance measures on convergence to fairness and efficiency

Description	*Value*
Packet size	1000 bytes
Maximum window	128 packets
TCP version	SACK
TCP timer granularity	0.1 seconds
RED queue limit Q	2.5 × B/W delay product
DropTail queue limit	1.5 × B/W delay product
RED parameters	min_{th}: 0.15Q, max_{th}: 0.5Q, w_q:0.002
	max_p:0.1, wait_ on, gentle_ on

Table 4: Network configuration

5 Simulation Results

We use the ns2 simulator[11] to validate that with RED [15] queue management, proposed controls, SIMD are TCP friendly and TCP compatible. In other words, we compare our controls to standard TCP [16], generalized AIMD [3], and IIAD [4] in terms of responsiveness, smoothness and aggressiveness. In most simulations, we include AIAD. In addition we investigate the way two same type of flows converge to their bandwidth fairness share and shows that SIMD algorithm out performs with other algorithms.

Unless explicitly specified in all of experiments, RED is used as queue management policy at bottleneck link. The bottleneck queue configurations and other simulation parameters are shows in Table 4.

The bottleneck queue size and RED queue attributes are tuned as recommended in[17]. The gentle option for RED queue is turned on as recommended in [18]. We choose β = 1/16 for SIMD and AIMD and $\alpha \approx$ 1/10 for AIMD to ensure for TCP friendliness. For IIAD, $\alpha = 1$ and $\beta = 2/3$. For AIAD, $\beta = 2/3$. For presentation, in the rest of this part, we call these implementations by their family name, like AIMD for AIMD(1/10,1/16) when there is no confusion. We use SACK [19] for congestion detection. We also obtained same results for other mechanisms such as Reno and newReno. We assume that there are no delayed acknowledgments.

6 Conclusions

To supply smoother transmission rate than that given by TCP, several TCP equivalent window based congestion control mechanisms have been suggested, including the general AIMD [1, 3] and TEAR [2]. These mechanisms use a moderate window step-down parameter to cut down rate variability, meanwhile using a matching window increment parameter to satisfy TCP friendliness.

We suggested and evaluated the first set of window supported TCP friendly congestion controls that use history selective information to improve transient behavior without sacrificing smoothness in steady state.

Additional approach to allow faster transmission rate is equation based congestion controls [5, 6, 7], first suggested in [59]. In these schemes, the end systems evaluate the packet loss rate and round trip time, and use TCP friendly equation [20] to calculate the transmission rate. Two comparisons [1, 9] of equation based and window based congestion controls have displayed that equation based schemes and window based AIMD share alike temporary behavior but equation based schemes allow for higher smoothness. However, the aggressiveness of equation based schemes is controlled by the nature of rate based control, which lacks a self clocking mechanism for overload protection as in window based control.

References:

[1] Sally Floyd, Mark Handley, and Jitendra Padhye, "A comparison of equation based and AIMD congestion control. http://www.aciri.org/floyd/chapters.html," May 2000.

[2] Injong Rhee, Volkan Ozdemir, and Yung Yi, "TEAR: TCP Emulation At Receivers – flow control for multimedia streaming," Tech. Rep., Department of Computer Science, North Carolina State University, April 2000.

[3] Y. Richard Yang and Simon S. Lam, "General AIMD congestion control," in Proceedings of ICNP, November 2000.

[4] Deepak Bansal and Hari Balakrishnan, "Binomial congestion control algorithms," in Proceedings of IEEE INFOCOM, April 2001.

[5] Sally Floyd, Mark Handley, Jitendra Padhye, and Joerg Widmer, "Equation based congestion control for unicast applications," in Proceedings of ACM SIGCOMM, August 2000.

[6] J. Padhye, J. Kurose, D. Towsley, and R. Koodli, "A model based TCP friendly rate control protocol," in Proceedings of International Workshop on Network and Operating System Support for Digital Audio and Video (NOSSDAV), June 1999.

[7] Wai Tian Tan and Avideh Zakhor, "Real time Internet video using error resilient scalable compression and TCP friendly transport protocol," IEEE Transactions Multimedia, vol. 1(2), pp. 172–186, June 1999.

[8] Sally Floyd and Kevin Fall, "Promoting the use of end to end congestion control in the Internet," IEEE/ACM Transactions on Networking, vol. 7(4), pp. 108–122, August 1999.

[9] Y. Richard Yang, Min Sik Kim, and Simon S. Lam, "Transient behavior of TCP friendly congestion control protocols," in Proceedings of IEEE INFOCOM, April 2001.

[10] Shudong Jin, Liang Guo, Ibrahim Matta, and Azer Bestavros, "TCP friendly SIMD congestion control and its convergence behavior," in Proceedings of IEEE ICNP, November 2001.

[11] E. Amir et al, "UCB/LBNL/VINT Network Simulator ns2" Available at http://http://www.isi.edu/nsnam/ns/.

[12] R. Puri, Kang Won Lee, K. Ramchandran, and V. Bharghavan, "An integrated source transcoding and congestion control paradigm for video streaming in the Internet," IEEE Transactions on Multimedia, vol. 3, no. 1, 2001.

[13] Shudong Jin, Liang Guo, Ibrahim Matta, and Azer Bestavros, "An Array of TCP friendly window based congestion control algorithms," Tech. Rep. BU CS 2001 015, Computer Science Department, Boston University, July 2001, Available at http://www.cs.bu.edu/techreports /2001 015 Array tcp friendly.ps.Z.

[14] Dah Ming Chiu and Raj Jain, "Analysis of the increase and decrease algorithms for congestion avoidance in computer networks," Computer Networks and ISDN Systems, vol. 17, pp. 1–14, 1989.

[15] Sally Floyd and Van Jacobson, "Random early detection gateways for congestion avoidance," IEEE/ACM Transactions on Networking, vol. 1(4), pp. 8–67, August 1993.

[16] Van Jacobson, "Congestion avoidance and control," in Proceedings of ACM SIGCOMM, August 1988.

[17] M. Christiansen, K. Jeffay, D. Ott, and F.D. Smith, "Tuning RED for Web Traffic," in Proceedings ACM SIGCOMM, Stockholm, Sweden, Aug. Sep. 2000.

[18] Sally Floyd, "Recommendation on using the "gentle" variant of RED," http://www.aciri.org/floyd/red/gentle.html, March 2000.

[19] M. Mathis, J. Mahdavi, S. Floyd, and A. Romanow, "TCP Selective Acknowledgement Options," Internet RFC 2018, April 1996.

[20] J. Padhye, V. Firoiu, D. Towsley, and J. Kurose, "Modeling TCP throughput: A simple model and its empirical validation," in Proceedings of ACM SIGCOMM, 1998.

About Author: Kulvinder Singh received the M.Tech.(CSE) degree in 2006 and the M.Phil.(CS) degree in 2008 from Ch. Devi Lal University Sirsa(Haryana), India. He is pursuing Ph.D. in Computer Science. At present he is working as Assistant Professor in Vaish College of Engineering, Rohtak, India. He is having more than 6 years of experience both in industry and academia. He is a member of IEC. He presents many research papers in national and international conferences. He published many research papers in various International Journals. His interest areas are Networking, Web Security, Internet Congestion and Fuzzy Database.