Asian Journal of Information Technology 5 (4) : 442-447, 2006

Effective Tcp Friendly Rate Control Scheme for Video Streaming over Wireless Networks

P. Yogesh, S. Bose and A. Kannan
Department of Computer Science and Engineering,
College of Engineering, Anna University, Chennai, India

Abstract: Rate control is an important issue in video streaming application for both wired and wireless networks. A widely accepted rate control method in wired networks is TCP Friendly Rate Control (TFRC). TFRC assumes that packet loss in wired networks is primarily due to congestion and as such is not be applicable to wireless networks in which the bulk of packet loss is at the physical layer. Hence the multiple TFRC Connections (MULTFRC) are being used as an existing approach, which is an end-to-end based solution to this problem. By opening appropriate number of TFRC connections, MULTFRC not only avoids modifications to the network infrastructure but also results in full utilization of the wireless bandwidth. However MULTFRC suffers form underutilization of resources and loss of granularity in number of connections. To address these drawbacks, we propose a modified TFRC rate control scheme that combines multiple connections to one connection and improves the performance of streaming over wireless networks. We name our rate control scheme as CTFRC (Consolidated TFRC) with loss handling mechanism. We carried out NS-2 simulations to compare the performance of MULTFRC and our own rate control scheme.

Key words: TCP friendly rate control, multiple TFRC, CTFRC, RTT

INTRODUCTION

Congestion control or rate control is an important issue for video streaming in both wired and wireless networks. If streaming rate is very low, it may lead to underutilization of the network bandwidth. On the other hand, too aggressive of a streaming rate could result in serious congestion collapse at the shared bottlenecks[1,2]. A widely accepted rate control method in wired networks is TCP friendly Rate Control (TFRC)[3]. TFRC makes use of TCP's throughput equation in which the rate is computed on the receiver side as a function of packet loss rate, Round Trip Time (RTT) and packet size[1]. This is well suited to video streaming and other multimedia applications.

TFRC is widely used for video streaming because of its stability, fairness to TCP flows and less fluctuation rate. Neither TCP nor TFRC can distinguish between packet loss due to buffer overflow and that due to bit errors[3,4]. Both of them treat any loss as a sign of congestion. TFRC underutilizes the wireless bandwidth due to the above reasons and its streaming throughput is less than that of the maximum throughput over the wireless. To overcome these limitations it was decided to make use of multiple simultaneous TFRC connections[4,5]. Hence this scheme is called the multiple TFRC scheme (MULTFRC) approach. This existing scheme is widely used for rate control in streaming over wireless. It performs without any modification to the network infrastructure or the protocols. It is based on the average queuing delay measurements where we go for increasing or decreasing the number of connections based on the queuing delay. The number of connections is optimized to achieve highest throughput and lowest packet loss rate. In spite of being considered as effective scheme, MULTFRC too suffers from limitations like excessive consumption of resources and utilization.

To address these limitations we propose a modified rate control scheme that makes use of the better side of TFRC and MULTFRC approaches. Our proposed scheme condenses the multiple connections into one connection, which performs without any modifications to the network infrastructure or the protocols and is discussed in detail in the following sections.

Floyd, S. *et al.* discusses the negative impact of increasing deployment of non-congestion-controlled best-effort traffic in the Internet[1,2]. Tan *et al.* proposes a point to point real-time video transmission scheme over the Internet that combines a low-delay TCP friendly transport protocol in conjunction with a novel compression method that is error resilient and bandwidth-scalable[3,5]. Yang *et al.* proposes a method that makes use

Coresponding Author: P. Yogesh, Department of Computer Science and Engineering, College of Engineering, Anna University, Chennai, India

of multiple simultaneous TFRC connections to overcome the limitation of underutilization caused by a single TFRC Connection[4]. There have been number of efforts to improve the performance of TCP or TFRC over wireless[6-8]. Snoop is a well-known TCP-AWARE local retransmission link layer approach[5]. A snoop resides on the router or base station on the last hop wireless link and records a copy of every forwarded packet. Assuming snoop module can access TCP acknowledgment packets (ACK) from the TCP receiver, it looks into the ACK packets and carries out local retransmission when a packet is corrupted by wireless channel errors. End-to-End statistics can also be used to detect congestion when a packet is lost[6-9]. Cen *et al.* presents an end-to-end based approach to facilitate streaming over wireless[10]. They combine packet inter-arrival time and relative one-way delay to differentiate between packet loss caused by congestion and that due to wireless channel errors. Yang *et al.*[4] proposes a cross-layer scheme that decides whether a packet loss is caused by wireless channel error or congestion, assuming only the last link is wireless. Tan *et al.* have proposed that idea of using dummy packets to actively probe the network, so as to differentiate between packet loss due to congestion and that due to channel error[5].

Explicit Loss Notification (ELN) can also be applied to notify the TCP sender when a packet loss is caused by wireless channel errors rather that congestion[11]. Other schemes such as[6-9] that use the end-to-end statistics to detect congestion can also be combined with TFRC to achieve rate control. The congestion detection scheme can be used to determine whether or not as observed packet loss is caused by congestion; TFRC can then take in to account only those packet losses caused congestion when adjusting the sending rate. Other similar works, but related to this approach include Multiple TCP (MULTCP)[12], which opens multiple connections to increase throughput. MULTCP was originally used to provide differential service and was later used to improve the performance in high bandwidth-round-trip-time product networks.

In this research, we propose a modified rate control scheme that integrates multiple connections into a single connection. Our proposed scheme (CTFRC with loss handling) includes loss detection and loss handler modules through which receiver and sender are able to cooperate and achieve optimum performance.

PROBLEM DESCRIPTION

The typical scenario that has been considered for streaming over wireless is shown in Fig. 1. where the

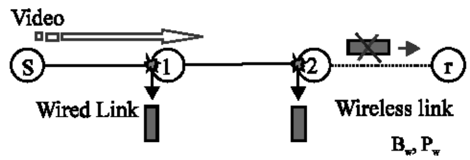

Fig. 1: Typical scenario for streaming over wireless

sender is denoted by *s* and the receiver by r. As shown the video server in the wired network is streaming video to a receiver in the wireless network, which is the last hop. The wireless link is assumed to have a specific bandwidth Bw and packet loss rate Pw, caused by wireless channel error or bit errors. This assumes that the maximum throughput over the wireless link is Bw (1-Pw).

There are many reasons for choosing this scenario. First: It is a simplified version of the popular cellular wireless data transmission scenario in which we are interested. Second: It captures the fundamental problem that we want to analyze. The scenario explains the fact that if there is going to be any packet loss in any of the intermediate nodes or even any error that is taking place at the wireless link the packets are not going to reach the receiver and we will not be able to receive the video with desired quality. However the TFRC as such is not applicable for streaming video over wired to wireless networks since it seems to underutilize the wireless bandwidth because of the streaming throughput i.e., T (1-p) is found to be less than the maximum throughput at the wireless link i.e., Bw (1-Pw). The wireless channel that is underutilized is referred as T (1-p)<Bw (1-p) where p is the end-to-end packet loss rate observed by receiver. Therfore this form of underutilization leads to reduction in the throughput performance and cannot effectively distinguish between the kinds of packet loss. Hence it was thought that without distinguishing between the kinds of packet loss whether it is due to congestion or due to channel or bit errors at the wireless, it is possible to pursue full utilization of the wireless bandwidth and lower packet loss rate by using simultaneous multiple TFRC connections thereby achieving higher throughput performance. Based on this scenario the two goals of the rate control are stated as follows: First, the streaming rate should not cause any network stability, i.e., congestion collapse. Second, it should lead to the optimum performance, ie. it should result in highest possible throughput and lowest packet loss rate. Therefore the receiver should apply some sort of congestion control scheme to compensate to the lost packets, which is the major working principle of the MULTFRC scheme. Here the total throughput of the application is expected to increase with the number of connections until it reaches

the hard limit of Bw (1-Pw). Moreover the wireless link is assumed to be fully utilized if T (1-P) = Bw (1-Pw), but round trip time is no longer at the minimum and overall packet loss rate would exceed the minimum limits.

The MULTFRC scheme makes use of the average queuing delay measurement to open optimum number of connections as explained in detail in section 4. In spite of being considered as an effective approach it suffers from certain demerits, as it is a multiple connection approach and also the quantization effects, which degrades, it's performance. Therfore in order to address these problems that are faced in the existing MULTFRC approach[3] the proposed approach with its framework is presented in section 4. It is basically a one-connection approach in contrast to the MULTFRC. Hence it eliminates the problems that are faced in the existing scheme. It is achieved by integrating the control law for the number of connections in MULTFRC into one connection with the same utilization performance as that of MULTFRC.

The related work[13] has proposed improvements to MULTFRC, the cost of modifying TFRC protocol. It follows the same strategy as MULTFRC in that it controls the way TFRC computes loss event rate and hence controls the sending rate accordingly. However, this research is also a multiple connection approach and as such suffers from quantization effect; and is more complex. Hence due the above reasons we propose the modified scheme of TFRC as an alternative to MULTFRC scheme.

PROPOSED SCHEME

In this section, we propose an alternative to MULTFRC, called Consolidated TFRC (CTFRC) with loss handling to address the above drawbacks, while retaining the same utilization performance of MULTFRC. It is basically an idea of how MULTFRC would behave like one TFRC like connection. To achieve this we integrate the number of connections in MULTFRC in to one TFRC connection. Moreover the loss detection and loss handler modules are integrated along with the framework of the new scheme as shown in the Fig. 2. Basically, the loss detection module at the receiver feeds back the RTT and loss event rate to the sender. The sender then adjusts sending rate to n times to that of one TFRC's sending rate. The functionalities of the CTFRC at sender and receiver are described as in Fig. 2.

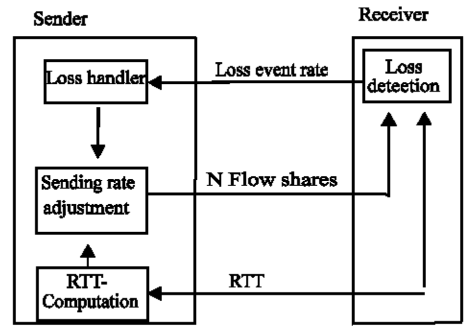

Fig. 2: CTFRC framework

Sender: The sender behaves as the receiver side when it receives feedback from the receiver. It adds every receiving packet to the packet history, calculates the sample RTT and averaging these RTT samples and calculates the loss event rate. When it behaves as a sender side itself, it adjusts the sending rate, sends the next packet, adds the packet-to-packet history and schedules the timer to send the next packet to the receiver. It then updates the sending rate n times that of one TFRC and removes the old packets from the history to store the new set of packets. The packets are sent to the receiver after adjusting the sending rate as a virtual single TFRC flow. The sender requires the loss event rate and the RTT values because it is with the help of these values only the throughput is calculated and the sending rate is adjusted accordingly.

Receiver: The receiver performs the loss detection where it marks every packet that appears as a lost packet. The marking of packets is the ECN (Explicit Congestion Notification), which is an indication of a congestion event. This ECN bit being generated for every lost packet is different from that of the sequence number of the packet, to differentiate the packet that is received and that is being lost at the receiver. The receiver also measures and updates the loss event rate based on the virtual TFRC flow with marked packets. The throughput is also calculated at the receiver side based on the packets that are received or those that are lost and is updated and fed to the sender when the receiver behaves as a sender.

On the whole the MULTFRC achieves n TFRC flow shares by opening n independent TFRC connections, while CTFRC achieves n flow shares by opening one connection and sending at n times the sending rate of one TFRC connection.

Hence the proposed approach eliminates the drawbacks of the existing MULTFRC approach because it is a single connection scheme and therefore it eliminates the need of operating with multiple connections where it demands an overhead of high computation overhead and space overhead. It also shows a better throughput and avoids the quantization step thereby reducing the implementation complexity of the simultaneous multiple TFRC connections (MULTFRC).

IMPLEMENTATION

The working performance of the MULTFRC approach makes use of the average queuing delay measurements. The fundamental idea is to increase the number of connections if the average queuing delay is less than threshold and to decrease it otherwise. The algorithm below explains the performance of the MULTFRC:

Algorithm 1-MULTFRC

- The initial numbers of connections are saved.
- Then increase the number of connections if the numbers of connections are less than the maximum number of connections using the preset parameter alpha i.e., 1.
- Updations of average throughput, RTT, ave-min-delay, gain on threshold.
- Make decision whether to increase/decrease connections based on the lost bits at the receiver.
- If the ave-RTT is lesser than ave-min-delay using the preset parameter then we increase the number of connections else we decrease the number of connections using the preset parameter beta i.e.,1.

Hence the above algorithm makes use of the average queuing delay measurements to increase or decrease the number of connections appropriately to open optimum number of connections thereby adjusting the sending rate at the sender side in each connections being opened and performs the effective rate control and utilizes wireless bandwidth efficiently.

Algorithm 2-CTFRC sender side

- It adds every sent packet in to the sent packet history and schedules the timer to send the next packet.
- Based on the feed back from the receiver it calculates the average RTT and updates the loss event rates and calculates the same. It then calculates the throughput and updates the sending rate.
- Scheduling the packets being sent by making use of the inter packet arrival time and the nominal time to sent the next packet.
- Performing the process to print the packet send and trim the send history.

Algorithm 3-CTFRC receiver side

- Checks the sequence number of packets received.
- Checks the ECN (Explicit Congestion Notification), which is marking of packets, which is an indication of packet loss.
- Estimates the loss and throughput based on the losses.
- Reports the loss and adjusts the sending rate.

Using the above algorithms both the MULTFRC and the CTFRC are simulated using the NS2 simulator. These simulations are carried out to show a how MULTFRC performs in terms of throughput, packet loss rate and the number of connections, all as a function of wireless channel error rate, b how CTFRC performs in terms end-to-end round trip time and throughput, as a function of wireless channel error rate. The topology used in simulations to show the performance of MULTFRC and CTFRC is as shown Fig. 3.

The sender is denoted as S and the receiver is denoted as the R. They both run the respective schemes at the application layer. For simulations the bottleneck link is assumed to have a specific bandwidth Bw. The bottleneck link is modeled by an exponential random packet loss model and wireless packet loss rate varies from 0.0 to 0.05. Actually, here the wireless channel error rate are artificially changed to see how MULTFRC and the CTFRC adapts to the change in error rate which is some sort of disturbances that is brought in to simulation to analyze their performance. Drop tail type of queue is used for each node. The throughput, end-to-end packet loss rate and number of connections opened of the MULTFRC system performances are shown in the Fig. 4.

Even in CTFRC the same topology is considered as it has the same working performance, as that of MULTFRC. And the bottleneck link is considered as the wired link with exponential random packet loss model same as that

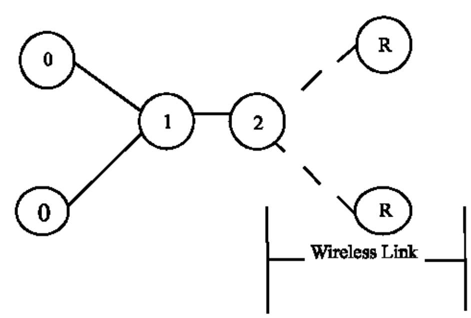

Fig. 3: The simulation topology used for both MULTFRC and CTFRC, all with a Bandwidth-1 Mbps and last hop as wireless

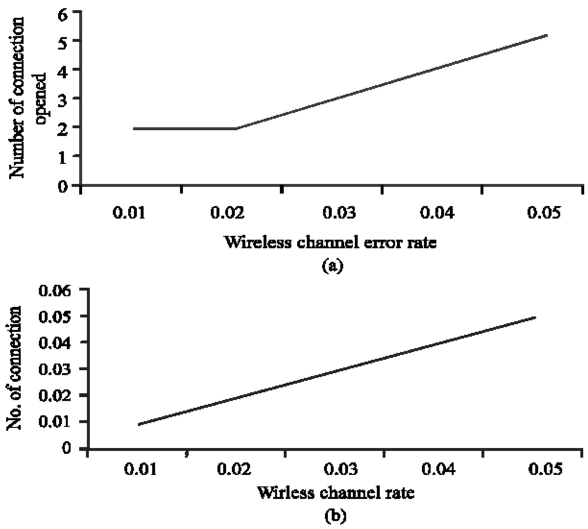

Fig. 4: NS2 simulations for Bw = 1Mbps (a) number of connections (b) end-to-end packet loss rate

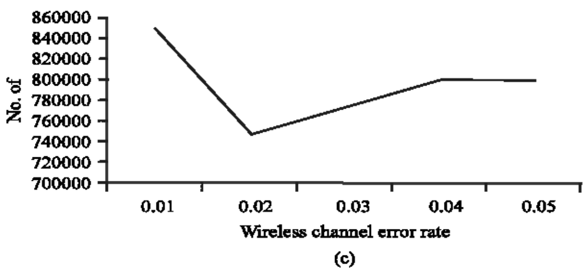

Fig. 4: NS2 simulations for Bw = 1Mbps (c) throughput, all as a function of packet error rate on wireless channel

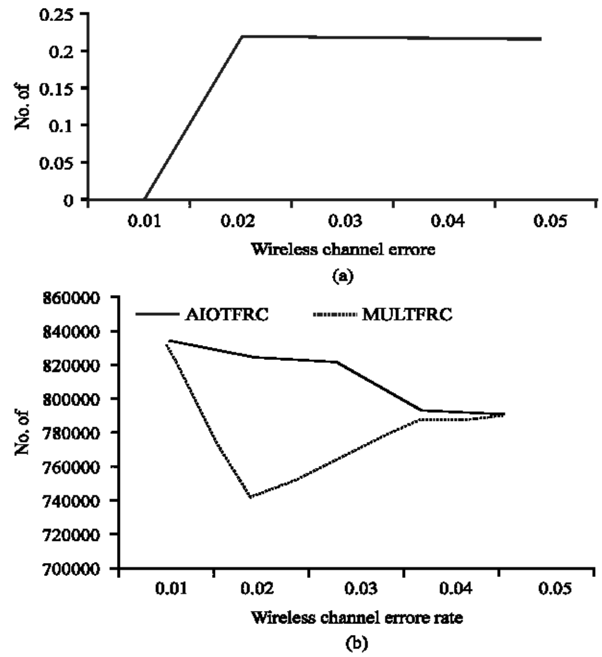

Fig. 5: NS2 simulations for Bw = 1 Mbps (a) End-to-end round trip time (b) throughput, all as a function of packet loss rate on wireless channel

of MULTFRC approach. The throughput performance and end-to-end roundtrip time performances of CTFRC are shown in the Fig. 5.

The above trace output explains the performance of the MULTFRC approach and the proposed CTFRC approach using NS2 simulations. The observation that is seen in Fig. 4a is the error rate is directly proportional to the number of connections that are opened. It means that if more connections are opened than the optimum number of connections, it leads to more of packet losses and to a larger end-to-end round trip time. Therefore it results in larger delay to recover the packet losses because the larger packet loss exceeds the lower bound. Moreover the strategy to be met by MULTFRC to open optimum number of connections fails. The optimum number of connections is important because that is the place where we can safely go for recovering the packet loss and where we can achieve the highest throughput. As seen in Fig. 4b where it is evident that as error rate increases there is more of end-to-end packet loss rate that is reason why optimum number of connections are opened to recover these high packet losses at highest throughput, which is depicted in Fig. 4c.

It is seen in Fig. 4c that the error rate is inversely proportional to the throughput. Initially the throughput is very high when there is only one connection, which is open. But the basic functionality is that as the number of connection increases the throughput increases up to a point, after which there is saturation. Therefore it is seen in Fig. 4c that at an error rate of 0.02 throughput decreases, so it is clear that at this point the optimum number of connection is not opened to safely recover the lost packets as the throughput performance at this point is very poor. But at the point 0.04 there is higher throughput performance after which there is saturation. Hence it is clear that at this point only an optimum number of connections of about 4 (Fig. 4a) is opened which shows a higher throughput performance with a packet loss rate, which can be recovered safely (Fig. 4 c). The reason for obtaining highest throughput performance is that only with the higher throughput performance there is better utilization of wireless bandwidth provided opening optimum number of connections and hence the strategy is met which is stated as," keep increasing the number of connections until an additional connection results in increase of end-to-end round trip time or packet loss rate.

To examine the dynamics of the CTFRC approach the performance metrics of throughout and end-to-end round trip time all as a function of packet loss rate in the wireless channel are shown in the Fig. 5a and Fig. 5b. As seen in Fig. 5b compares the performance of MULTFRC and

CTFRC. It shows that the proposed approach seems to outperform the existing MULTFRC scheme. At an error rate of 0.02 even though the MULTFRC shows a bigger drop in the throughput the performance CTFRC is much better as its throughput is a much higher value than that of the MULTFRC scheme thereby leading to a better utilization wireless bandwidth. Moreover the throughput significantly outperforms the MULTFRC even at higher error rate and better throughput even at lower error rate scenarios. This can be seen from the Fig. 5b at the same error rate of 0.04 the performance of CTFRC is a much higher value than that of the MULTFRC that is clearly depicted. As far as the end-to-end round trip time is concerned the performance is uniform as the CTFRC is a one-connection approach and hence does not lead to a much variation in the end-to-end round trip time and hence would be the minimum of end-to-end round trip time.

Therefore it achieves greater wireless bandwidth utilization with better throughput and readily avoids the quantization effect and reduces the implementation difficulty that was seen in MULTFRC.

CONCLUSION

Thus in this study, the existing end-to-end rate control scheme for wireless video streaming has been examined which achieves both high throughput and low packet loss rate, without having to modify the network infrastructure or protocols. The existing scheme is based on increasing the number of connections thereby opening optimum number of connections to enhance higher utilization of the wireless bandwidth. In spite of being considered as an effective scheme it suffers from the quantization effect and the overhead of operating multiple connections. Therfore in order to address these two issues the CTFRC has been proposed which achieves these goals by creating one connection whose throughput performance is much higher that that of the MULTFRC even at same and higher error rates. It does so by measuring the round trip times, adjusting the sending rates to recover the lost packets and to obtain the required output. NS2 simulations show that CTFRC achieves better throughput than that of MULTFRC even at higher packet loss rates. The CTFRC is also fair to TCP flows because it directly makes use of the TCP's throughput equation to adjust its sending rates based on the packet loss rates.

In this study we don't consider the allocation of resources on behalf of the connections. The future work may include allocating proper resources in order to achieve the perceived video quality at the receiver. Since wireless is a dynamic environment, rate control based on RTT measurements leads dramatic fluctuations in the network bandwidth and delay and hence the end result is not accurate and may have fluctuations. Hence the appropriate resource allocation is to be done in order to compensate the vast variations of the dynamic wireless environment and to achieve the perceived video quality.

REFERENCES

1. Floyd, S., H. Mark, P. Jitendra and W. Joerg, 2000. Equation-based congestion control for unicast applications, Proc. ACM SIGCOMM 2000, pp: 43-56
2. Floyd, S. and K. Fall, 1999. Promoting the use of end-to-end congestion control in the internet, IEEE/ACM Transactions on Networking, 7: 458-472.
3. Chen, M. and A. Zakhor, 2005. Rate control for streaming video over wireless, IEEE Wireless Commun., 12: 32-41.
4. Yang, F., Q. Zhang, W. Zhu and Ya-Qin Zhang, 0000. An end-to-end tcp-friendly streaming protocol and bit allocation for scalable video over wireless internet, IEEE J. Selected Areas in Commun., 22: 779-790.
5. Tan, W. and A. Zakhor, 1999. Real-time internet video using error resilient scalable compression and tcp-friendly transport protocol, IEEE Transactions on Multimedia, 1: 172-186.
6. Balakrishnan, H., V. Padmanabhan, S. Seshan and R. Katz, 1996. A comparison of mechanisms for improving TCP Performance over Wireless links, *ACM SIGCOMM'96*, pp: 256-269.
7. Biaz, S. and N.H. Vaidya, 1999. Discriminating congestion loss from wireless losses using inter-arrival times at the receiver, Proc. of IEEE Symposium on Application-specific System and Software Engin. Tech. Richardson, TX, pp: 10-17.
8. Samaraweera, N., 1999. Non-congestion Packet Loss detection for TCP Error Recovery using Wireless links, in IEE Proce. Commun., 146: 222-230.
9. Tobe, Y., Y. Tamura, A. Molano, S. Ghosh and H. Tokuda, 2000. Achieving moderate fairness for udp flows by path status classification, In Proc. 25th Ann. IEEE Conf. on Local Computer Networks (LCN 2000), Tampa, FL, pp: 252-261.
10. Cen, S., P.C. Cosman and G.M. Voelker, 2002. End-to-end differentiation of congestion and wireless losses, Proc. Multimedia Computing and Networking (MMCN) conf. San Jose, CA, pp: 1-15.
11. Balakrishnan, H. and R. Katz, 1998. Explicit loss notification and wireless web performance, IEEE Globecom Internet Mini-Conference.
12. Croft, J.C. and O. Philippe, 1998. Differentiated end to end internet services using a weighted proportional fair sharing TCP, ACM Comput. Commun. Rev., 28: 53-69.
13. Tong, X. and Q. Huang, 2005. Multfrc-lerd: An improved rate control scheme for video streaming over wireless, in Proc. 5th Pacific Rim Conference on Multimedia, Tokyo, Japan, pp: 282-289.