Proceedings of the 6th WSEAS International Conference on Multimedia Systems & Signal Processing, Hangzhou, China, April 16-18, 2006 (pp168-172)

# Dynamic Flow Control Method using Relative Send Trip Time

Ha-sung Koo, Ho-keun Song

ABSTRACT

This paper proposed a new dynamic flow control algorithm for teleconferencing systems that has advantages over previous methods. Our method makes use of Relatively Send Trip Time(RSTT), and passes through two stages to measure the network status. In the first stage, we categorize the network status into three different states depending on difference between current RSTT and previous RSTT. In the second stage, our method readjusts the states of network according to the predefined RPL. Depending on the network status, we estimate the possible use of bandwidth of the current network. For this purpose, transmission control ratio is calculated by applying the different weight factors to the current network status, previous network status and the network status before last, respectively. And then the ratio is applied to current transmission rate. In the experiments, we measure the performance of the proposed method and compare the results with those of the previous methods. According to the results, our method is proven to show improved RPL and an increased total number of transmitted packets over a given period time than the other methods.

*Key-Words: network flow control, packet loss, round trip time, bandwidth control,*

## 1 Introduction

More and more applications and services need to transmit multimedia data over increasingly congested networks. One such application is the video teleconferencing system in which video data needs to be streamed to multiple recipients over continuously changing networks. This type of video teleconferencing system must be able to quickly adapt to changing network status. However, the existing Transmission Control Protocol (TCP) and User Datagram Protocol (UDP) technologies do not provide sufficient bandwidth capacities for this system. To improve the video quality for end users, various end-to-end flow control methods have been proposed [1]-[5]. An end-to-end flow control method estimates 6the possible use of bandwidth in the current network status and then obtains the transmission rate that it controls.

One such method uses the Rate of Packet Loss (RPL) to determine the status of the current network[3]. RPL is a reliable measure for revealing congested network status, but the RPL method is problematic in that it increases network traffic by raising the transmission rate until packet loss is generated.

Another method makes use of the Round Trip Time (RTT)[5] to adjust the transmission rate of the network. RTT refers to the time it takes for a packet to move, round trip, between the sender and the recipient in the video teleconferencing system. The RTT can quickly and easily measure the current network status before packet loss occurs. However, when packet loss is caused by a poorly performing recipient system or is due to an overload of network traffic, the RTT method cannot cope with the rapid loss of packets.

A hybrid method[6]is proposed that uses both the RPL and the RTT. It is a TCP-like flow control algorithm. In the method, when the RPL is under normal conditions, current network status falls into one of three defined states by using the averaged RTT. If the RPL exceeds the critical level of condition, the network is deemed to be in a congested state. Depending on the estimated network status, the method measures the possible use of the bandwidth and obtains the suitable transmission rate of the network. However the method has some erroneous estimation of the network status, because it only compares currently reported RTT with previously measured RTT.

Therefore, this paper proposes an end-to-end dynamic flow control algorithm that employs Relative Send Trip Time (RSTT).

## 2 Proposed Dynamic Flow Control Algorithm

### 2.1 Measuring Relatively Send Trip Time

Our method makes use of Relative Send Trip Time(RSTT) to adjust the transmission rate of the network. The RSTT refers to the relative time it takes for a packet to round trip between the sender and the recipient in the teleconferencing system.

In a video teleconferencing system that uses the Real time Transport Control Protocol (RTCP), LSR (Last Sender Report) and DLSR (Delayed since Last SR) information of the SR (Sender Report) and RR (Receiver Report) packet can be used to measure the round-trip time between the sender and recipient. When the sender transmits the SR packet, the transmission time is recorded in the LSR of the SR packet. At the other end, when the recipient sends back the RR packet in response to the SR packet, the LSR of the RR packet is copied from the LSR of the SR packet. In addition, the recipient records the delay time, DLSR, between the time that the SR packet arrived and the time that the RR packet was sent. Therefore, the RSTT between the sender and the recipient can be calculated using the following equation:

RSTT = (LastLSR-CurLSR)-(LastDLSR- CurDLSR)

Fig. 1 represents the relationship of the equation where (x-y) is the RSTT. If the RSTT is positive value, then the network status has worsened. On the contrary, if the RSTT is negative value, then the status has improved.

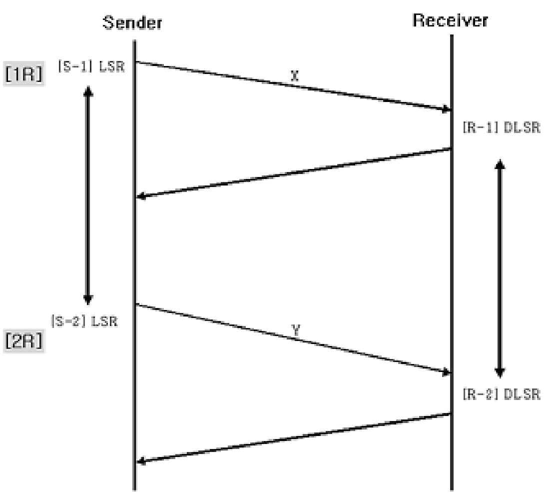



Fig. 1 Relationship of some variables for calculating the RSTT.

### 2.2 Measuring Network Status

A pseudo-code for measuring network status whenever the RR packet of the RTCP arrives at the sender is shown in Fig. 2. The proposed method passes through two stages to measure the network status.

```
(1)  LONG lCurSTTime = 1000 - ((lCurSRTime - lLastSRTime) - (lCurRRTime - lLastRRTime));

(2)  if(lCurSTTime < lLastSTTime)
     {
(3)      dCurIncreaseFlat = NET_WINDOW_INCREASE;
     }
(4)  else if(lCurSTTime == lLastSTTime)
     {
(5)      dCurIncreaseFlat = NET_WINDOW_MAINTAIN;
     }
(6)  else
     {
(7)      dCurIncreaseFlat = NET_WINDOW_DECREASE;
     }


(8)  if(lPacketLossRate)
     {
(9)      if( dCurIncreaseFlat <= 0)
         {
(10)         dCurIncreaseFlat  -=  0.15;
         }
(11)     else
         {
(12)         dCurIncreaseFlat = NET_WINDOW_CONGESTED;
         }
     }
```

**Fig. 2.** Proposed Network Status Measurement Algorithm

In the first stage, as shown in lines (2) through (7) of Fig. 2, our algorithm categorizes the network status into three different states. The three states are *NW_Increase*, *NW_Maintain*, and *NW_Decrease*. Each state means that network bandwidth can be increased, maintained or decreased, respectively.

The network status falls into one of the three states depending on difference between current RSTT and previous RSTT. If current RSTT is smaller than previous RSTT, then current network status has improved. Therefore the network status is defined as *NW_Increase*, *etc*.

In the second stage, as shown in lines (8) through (12) of Fig. 2, our method readjusts the states of network according to the predefined RPL. In lines (8) through (10), if the network traffic has some loss of packets but the network status, defined in the first stage, is *NW_increase* or *NW_Maintain*, then we assume the packet loss is caused by a poorly performing recipient system or by transient network traffic overload. And then we perform downward adjustment of the network status by one level. Otherwise, if the network traffic has some loss of packets and the network status is predefined as *NW_Decrease*, then we assume that the network is deemed to be in a congested state. Therefore the network status is redefined as the fourth status, *NW_Congested*.

### 2.3 Adjustment of Transmission Rate

Depending on the status of the network, we estimate possible uses of bandwidth and obtain suitable transmission rates of the network. For this purpose, our method controls the transmission rate by using not only the current network status but also the past status as shown in Fig. 3.

```
(1) dFrameControlRatio = (dCurIncreaseFlat * 0.5) + (dLastIncreaseFlat * 0.35) + (dDLastIncreaseFlat * 0.15);
(2) dLastFrame        += dLastFrame * dFrameControlRatio;
```

Fig. 3 Transmission rate control algorithm

In line (1), transmission control ratio is calculated by applying the different weight factors to the current network status, previous network status and the network status before last, respectively. On the other hand, each control ratios for the network status are defined as Fig. 4 and are selected based on experimental data.

```
#define NET_WINDOW_INCREASE     0.2
#define NET_WINDOW_MAINTAIN     0.05
#define NET_WINDOW_DECREASE     -0.1
#define NET_WINDOW_CONGESTED    -0.2
```

Fig. 4 Control ratios according to the network status

And then the transmission control ratio is applied to current transmission rate as in line (2). As a result, a new transmission rate for the current network is determined.

## 3 Experiment and Results

In this section, we present experimental results of the proposed flow control algorithm.

### 3.1 Experiment environment

The testing platform is a Pentium III PC running at 850MHz. The platform used a video conferencing tool [2], VIC 2.8, to measure the quantity of transmissions and packet loss on the conferencing network path. The experiment network path between Hanseo University and Edmonton in Canada includes a total of 14 routers as in Fig. 5.

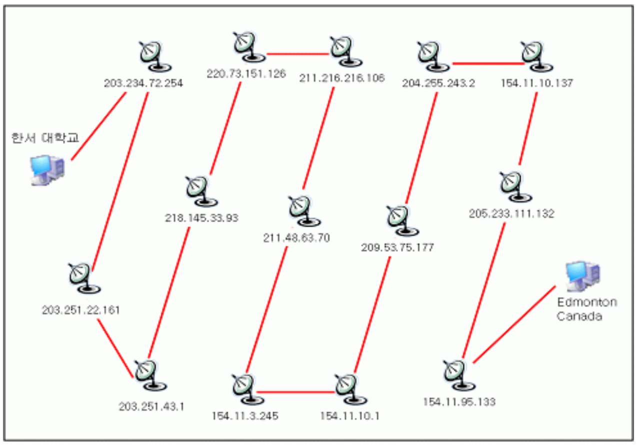



Fig. 5 Experiment network path

### 3.2 Measuring the performance

To measure the actual bandwidth of the network between the two organizations, a program called PING was developed using the echo of ICMP (Internet Control Message Protocol) [7]. For our purposes, the PING's ICMP echo packets were generated every 5 seconds as in RTCP control.

Fig. 6 shows the interface of the simulation program that is designed for measuring the performance of the proposed method. The program displays some information about amount of transmitted packets, current network status, and flow pattern of network, etc.

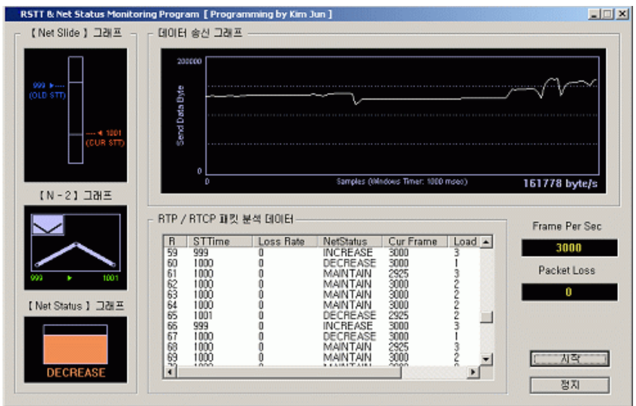
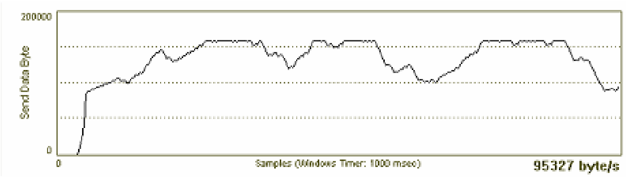
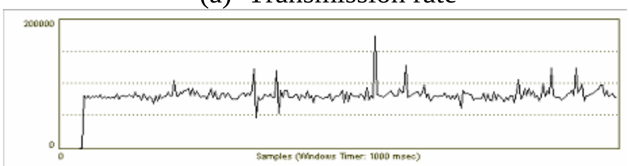



Fig. 6 Simulation program

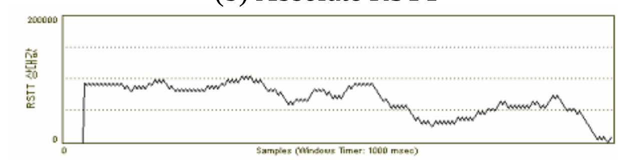
Fig. 7 depicts the performance of the proposed method in terms of transmission rate. In the graph, we can find some points on the rapid decrease of the transmission rate, when the network falls into successive status of *NW_Decrease* and has some overload of network traffic which cause packet loss. If the network traffic has some loss of packets but the network status is *NW_increase* or *NW_Maintain*, then the transmission rate is decreased slightly. In the final analysis, the graph reveals that our method maintains its stability of the transmission rate over continuousely changing network.




(a) Transmission rate




(b) Absolute RSTT




(c) Relative RSTT

Fig. 7 Performance graph of the proposed method

### 3.3 Comparison with the previous methods

In this section, we compare the performance of our proposed dynamic flow control algorithm with those of the previous methods[3,5,6].

As mentioned, the RPL method is problematic in that it increases the network traffic by linearly raising the transmission rate until packet loss is generated, and then geometrically decreasing the rate. As a result, the transmission rate control repeats up-and-down and has nothing to do with the current network status. On the contrary, the RTT method can adjust the transmission rate until packet loss occurs. However, the RTT method cannot cope with the rapid packet loss because it disregards the RPL in determining network status. On the top of those methods, A hybrid method[6], a TCP-like flow control algorithm using RTT as well as RPL, shows a better performance than those of the previous methods. However the method also has some erroneous estimation of the network status, because it only compares currently reported RTT with previously measured RTT.

In our proposed method, it is not the network status of RTT itself but those of the RSTT patterns that control the transmission rate of the network. Table 1 shows a quantitative performance analysis of the methods. According to the results, our method is proven to show improved RPL and an increased total number of transmitted packets over a given period time.

Table 1 Comparision of packet loss

| | RPL | RTT | Hybrid | Proposed |
|---|---|---|---|---|
| Total number of Packets | 14,862 | 14,964 | 17,249 | 22,716 |
| Total number of Packet Loss | 783 | 221 | 118 | 32 |
| Rate of Packet Loss (%) | 5.268 | 1.476 | 0.684 | 0.140 |
| Operation Time (min.) | 40 | 40 | 40 | 40 |

## 4 Conclusion

This paper proposed a new dynamic flow control algorithm for teleconferencing systems that has advantages over previous methods. The RPL is a good measure for revealing congested networks, and the RTT method can adjust the transmission rate until packet loss occurs. A hybrid method, a TCP-like flow control algorithm, uses both the RPL and the RTT. The method reports a good result than the former methods. However the method has some erroneous estimation of the network status, because it only compares currently reported RTT with previously measured RTT.

Our method makes use of Relative Send Trip Time(RSTT), and passes through two stages to measure the network status. In the first stage, we categorize the network status into three different states depending on difference between current RSTT and previous RSTT. In the second stage, our method readjusts the states of network according to the predefined RPL. Depending on the network status, we estimate the possible use of bandwidth of the current network. For this purpose, transmission control ratio is calculated by applying the different weight factors to the current network status, previous network status and the network status before last, respectively. And then the ratio is applied to current transmission rate. In the experiments, we measure the performance of the proposed method and compare the results with those of the previous methods.

According to the results, our method is proven to show improved RPL and an increased total number of transmitted packets over a given period time than the other methods.

Future studies should consider more diverse network environments or conditions.

## References:

[1] Steven McCanne, Van Jacobson.: vic: a flexible framework for packet video. ACM Multimedia, Nov. (1995)

[2] V. Jacobson.: Congestion Avoidance and Control. Proceedings of ACM SIGCOMM '88, Stanford, Aug. (1988)

[3] Ingo Busse, Bernd Deffner, Henning schulzrinne.: Dynamic QoS Control of Multimedia Application based on RTP. http://www.fokus.gmd.de/step/acontrol/ac.html. May (1995)

[4]S.K.Nah, D.H.Ko, J.S.Ahn, S.B.Kim.: Comparison of two dynamic flow control algorithms for teleconferencing system. SoftExpo 97, Dec. (1997) 381-390

[5]K.D.Hwan, S.K.Nah, J.S.Ahn.: Network bandwidth estimation using RTCP for teleconferencing system. Proceedings of KISS autumn conference, Vol. 24, No. 3, Oct. (1997) 283 - 286,

[6] H.K.Song, H.S. Koo, "Dynamic End-to-End Flow Control for Teleconference System", WSEAS Transactions on Computers, Issue 12, Vol. 4, pp.1761-1767, Dec. 2005.

[7] J. Postel.: Internet Control Message Protocol - DAPRA Internet Program Protocol Specification. RFC792, Sep. (1981)