An IP DiffServ Framework For Real-time Video Transmission

Min Chen, Gang Wei, and Xiaosong Zhu
Dept. of Electronic and Communication Engineering, South China University of Technology
Email: chenminnew@21cn.com zhuxiaosong@21cn.com

***Abstract*—This paper presents an improved IP differentiated services (DiffServ) framework which is suitable for real-time video transmission. Especially, we investigate MPEG-4 video transmission. Based on the feature of MPEG-4 flow and the extent of network congestion, implements a new DiffServ marking algorithm. In the border router, we propose a new scheduling algorithm, called the Bit-based Weight time Slot Compensate (BWSC) scheduling algorithm, which dynamically adjust the weighted quantum of service. Under the condition that network bandwidth can't guarantee the demand of MPEG-4 real-time transmission, the PSNR comparison of restored MPEG-4 video frame sequence verifys our method has much more better performance than Best-effort network. End-to-end delay and delay jitter also meet the real-time transmission demand in the improved DiffServ network.**

Keywords—DiffServ; Scheduling algorithm; MPEG-4;

I. Introduction

With multimedia technology developing rapidly, traditional IP networks, which only provide best-effort services, can not meet requirement for the transmission of the convergence of data, voice and video traffic, especially for the real-time video transmission since packets would suffer the non-guaranteed delay, jitter and loss in this environment. To solve the problem, the DiffServ (Differentiated Services) approach is developed by the IETF (Internet Engineering Task Force). It try to provide quality of service in networks [1]. It classifies packets into one of a small number of aggregated classes, based on the DiffServ CodePoint (DSCP) located in the IP packet's header. At each DiffServ router, packets are subjected to a Per-Hop Behavior (PHB), which is defined to provide a specific forwarding behavior and invoked by the DSCP value. A small number of specific PHBs has been standardized by the DiffServ working group to provide differential treatment of traffic which are EF (Expedited Forwarding) PHB [2], AF (Assured Forwarding) PHB [3] and BE (Best Effort) PHB.

The topic of video transmission over DiffServ networks has been discussed in some papers [4~7], where proposed some video marking algorithm that marking the packets at the video server before transmission according to their contribution to the perceived picture quality. If the packet is marked as no important, then it will be dropped first in case of congestion. In [4][5], the base layer stream, the enhanced layer 1 stream and the enhance layer 2 stream are marked with the low drop precedence, medium drop precedence and high drop precedence respectively. In this paper, we implement a novel DiffServ video marking algorithm based on the combination of the transmission condition of network and the different importance of video streams for decoder over DiffServ networks.

To support different PHB, border router must deploy a DiffServ scheduler. In [8], a new service discipline called *Dynamic Weighted Round Robin (DWRR)* scheduling algorithm, which is based on WRR with introducing adaptive quantum of service varying traffic conditions, is proposed. It is suitable for real-time VBR service in ATM network, where the smallest transmission unit is a fixed size cell of 52 bytes, so it is not suited for use over IP networks. Paper [7] implement Weight Round Robin scheduling algorithm to support DiffServ video marking algorithm, using different fixed weight settings to decrease packet delay and loss rate of I frame and P frame. For real-time VBR traffic, we hope provide more service chance to reduce their ETE delay when burst arrives. And the additionally occupied service quantum will be compensated when burst leaves, if the reserved bandwidth exactly meets the average bandwidth that real-time VBR traffic requires. Based on the analysis above, this paper propose a Bit-based Weight time Slot Compensate (BWSC) scheduling algorithm, it is suitable for real-time VBR service over IP DiffServ networks. BWSC dynamically adjust quantum of service for real-time VBR burst traffic to reduce their ETE delay and jitter.

Especially, we focus on the scenarios with very low channel bandwidth which cannot provide enough bandwidth for the base layer stream transmission. When channel bandwidth is lower 20 percent than that of real-time MPEG-4 video transmission requires, video client still can restored picture effectively using error concealment technology under proposed network.

The rest of this paper is organized as follows. Section 2 depicts proposed IP DiffServ network, and describes our DiffServ video marking algorithm. Section 3 presents the scheduling strategy with a pseudo-code illustration of the BWSC scheduling algorithm. Simulation results and analysis to justify the significance of our schemes will be explained in Section 4. Finally, Section 5 will conclude the paper.

II. IP DiffServ Network Model

Figure 1 shows our IP DiffServ network model. It consists of a MPEG-4 video server, a FTP&Email server and a client which are connected respectively with the IP core network by border router. The FTP&Email server generates best-effort (BE) background traffic. The real-time video traffic streams are produced by the video server, and encapsulated into RTP/UDP/IP packets. The DSCP value is marked in the IP header. When packets are transmitted to the DiffServ border router, they will be classified, conditioned and scheduled. Different queue is allocated weighted bandwidth according to which packet class it serve. The core network will simply forward packets. Finally packets are routed to border router 2 and then to the client. The client will decode each frame before its deadline.

A. DiffServ marking algorithm in video server

There are three basic types of frames in the MPEG video stream: I frames, P frames and B frames. I frame is very important for decoding, and is necessary for the prediction coding of other frames. P frame has as similar importance as I frame. But if part of B frame is lost, the impairment propagates is only in that frame. In addition, to improve the resilience to the error occurred in the video stream, data partitioning is used in MPEG-4 to separate the MV (Motion Vector) from texture field. When errors are detected solely in the texture field, the MB (Macro Block) will be reconstructed using correct MV. So MV data of P frames is more important than texture data of P frames.

Based on the analysis above, a new algorithm is proposed to mark video stream with appropriate IP DiffServ PHB by four strategies as follows:

Let DP be the abbreviation of Drop Precedence.

Policy 1: if the bandwidth is enough, then
DSCP = AF Low DP, for all frames (e.g.: AF1)

Policy 2: if the bandwidth is insufficient, but more than 70% of required bandwidth, then
DSCP = AF Low DP, for I and P frames (e.g.: AF1)
DSCP = AF High DP, for B frames (e.g.: AF2)

Policy 3: if the bandwidth is insufficient seriously, and less than 70% of required bandwidth, then
DSCP = AF Low DP, for I frames and MV data of P frames (e.g.: AF1)
DSCP = AF High DP, for B frames and texture data of P frames (e.g.: AF2)

B. Implement border router

Complex traffic conditioning such as classification, marking, shaping, and policing are pushed to network border routers to make the functionalities of the core routers relatively simple. Therefore, implementing DiffServ border router is important. Our model has several key points as follows:

(a) Divide MPEG-4 data into AF1 and AF2 flows; (b) DiffServ router receive MPEG-4 video stream and BE traffic simultaneously; (c) Deploys two queues, one FIFO-queue for AF1 flows, one RIO-queue for AF2 and BE flows; (d) In the meter, different token mechanism is introduced to shape different flows before entering the queue; (e) Multi-class Random Early Detection (RED) is used in RIO-queue as the packet drop scheme to provide different RED parameters for AF2 and BE flows. The differentiation between AF2 and BE flow is achieved by setting different dropping threshold. The dropping threshold of BE flow is lowest, then the AF2; (f) The BWSC schedule scheme proposed in section three is adopted in scheduling different queues.

The Meter performs IN/OUT (in-profile / out-of-profile) checking on each incoming packet. We use two token buckets to check the conformance of AF packet, one for AF1 packet, the other for AF2 packet. If AF1 packet fails to pass the check, it will be dropped, which seldom happens since we set the token rate of AF1 Bucket large to guarantee its low drop rate. While AF2 packet fails to pass the check, it will be degraded and remarked as BE packet. The remarked AF2 packet will not suffer from discarding, but must compete equally with BE flow within RIO-queue, which make AF2 flow occupy bandwidth as possible as it can beyond reserved bandwidth.

The packet must satisfy two conditions in order to be served: 1) for AF1 packet, it must get enough tokens not to be dropped in the meter; 2) for AF2 or BE packet, it is not be dropped by RED mechanism in RIO-queue.

III. Bit-based Weight Time Slot Compensate Scheduling Algorithm

Now, we present the scheduling strategy with a pseudo-code illustration of the algorithm to provide rtVBR service effectively. In [8], a *Dynamic Weighted Round Robin (DWRR) scheduling discipline* is proposed, which is suitable for rtVBR service, but the smallest transmission unit is a cell whose size is fixed in ATM network. It is not suitable for variable-length packets in IP network. In [7], WRR with different weight settings is using to decrease the packet delay time and the packet loss rate of I and P frames. To further promote the ETE delay performance of rtVBR traffic, we propose BWSC scheduling scheme based on the analysis above. It dynamically adjusts weight to give more service chance to delay sensitive traffic.

In BWSC, we introduce a constant called *overdraft time threshold* (TH_i) which satisfy:

$$0 \le TH_i \le MaxPk\_service\_time \qquad (1)$$

Because the threshold is used to judge whether or not serve the last packet of one slot. If the $Pk\_service\_time$ at the top of the queue is larger than $Left\_service\_time_i$, BWSC will still dequeue the packet if the value that $Pk\_service\_time$ subtracts $Left\_service\_time_i$ is less than TH_i, but the overdraft service time must be compensated from the quantum for the next slot to avoid causing starvation of other queues.

Using TDMA for reference, BWSC divides time into time slot, each slot will serve one queue in turn. But the time quantum of slot is weighted and dynamically adjusted. Especially, we configure large TH_i in FIFO where I frame and P frame are enqueued. When I frame is encoded, there will be burst arriving in FIFO, the additional overdraft service time can alleviate effect of burst. When burst disappears, additional overdraft service time must be compensated to guarantee fairness to other queues.

Algorithm for Bit-based Weight time Slot Compensate

Constants:

N: number of queues/slot;

Q_i: queue i (i = 0,1,..., N-1);

W_i: weight allocated to Q_i in one slot;

w: time value per weight;

TH_i: overdraft time threshold of Q_i;

Variables:

$Weighted\_time_i$: weighted service time allocated to Q_i

$Service\_begin\_time_i$: the time begin to serve Q_i;

$Expected\_service\_end\_time_i$: service end time of Q_i according to weighted time allocated in one slot;

$Left\_service\_time_i$: Left service time for the service of Q_i in current slot;

$Overdraft\_time_i$: Overdraft service time for the service of Q_i;

$Service\_rate$: the serve rate of router (bps);

$Pk\_service\_time$: service time needed for packet;

Initialization:

For ($i = 0; i < N; i = i + 1$)

$Weighted\_time_i = w \times W_i$;

$Overdraft\_time_i = 0$;

En-queuing: [on arrival of packet $p(i)$ **entering** Q_i**]**

ENQUEUE (Q_i, $p(i)$);

Dequeuing:

$i = 0$;

While FOREVER do

$Service\_begin\_time_i = Current\_simulation\_time$;

$Expected\_service\_end\_time_i = Service\_begin\_time_i + Weighted\_time_i - Overdraft\_time_i$

$Left\_service\_time_i = Expected\_service\_end\_time_i - Service\_begin\_time_i$

While (Q_i not empty) do

$PacketSize$ = Size (Head (Q_i));

$Pk\_service\_time = PacketSize / Service\_rate$;

If ($Pk\_service\_time \le Left\_service\_time_i$)

then

Send (Dequeue(Q_i));

$Left\_service\_time_i = Left\_service\_time_i - Pk\_service\_time$

($Left\_service\_time_i \ge 0$)

Else if (($Pk\_service\_time - Left\_service\_time_i$) $< TH_i$) **then**

Send (Dequeue(Q_i));

$Overdraft\_time_i = Pk\_service\_time - Left\_service\_time_i$

($Overdraft\_time_i < 0$)

break; (*end serve this queue in current slot*)

Else

$Overdraft\_time_i = Pk\_service\_time - Left\_service\_time_i$

($Overdraft\_time_i < 0$)

(*end serve this queue in current slot*)

End of While

If (Empty(Q_i)) then

$i = (i + 1)\ Modulo\ N$; (*change slot*)

End of While

IV. SIMULATION AND ANALYSIS

Using the Diffserv network simulation model mentioned in section 2, we have done numerical experiments on video encoding, transmission and decoding. The border router parameter settings are showed in Table I. The video sequence is Foreman. Under both DiffServ and best-effort network, we compare the performance of PSNR of reconstructed picture. The data partitioning technology is used for encoding. The environment of experiment is as follows: frame rate is 30fps, average bit rate of source encoding is 80kbps, average bit rate after encapsulated is 98kbps. BE traffic bit rate is 30kbps. The channel bandwidth is 80kbps, and the IP packets are marked using policy 2 according to the DiffServ marking algorithm. The comparison for the PSNR of the two cases is showed in Figure 2. It can be seen, with DiffServ, the value of PSNR keep stable on the whole and the replay of reconstructed video sequence is clear and smooth. While with best-effort, the average value of PSNR decreased rapidly and is 11.67dB less than that with DiffServ, and the subjective quality is poor.

The curves of ETE delay with DiffServ are showed in Figure 3, the average delay is lower than 50ms. And Figure 4 show the delay jitter is about 3ms. So the delay and jitter meet the requirement for real-time video transmission.

TABLE I. BORDER ROUTER PARAMETERS SETTINGS

Border Router Parameters Settings	DiffServ	Best-effort
FIFO-queue length(bit)	2,000	10,000
RIO-queue Weight	100	—
FIFO-queue length(bit)	2,000	10,000
RIO-queue Weight	50	—
Time value per weight(s)	0.001	—
Service rate (bps)	80,000	80,000
Overdraft time threshold(s)	0.003	—

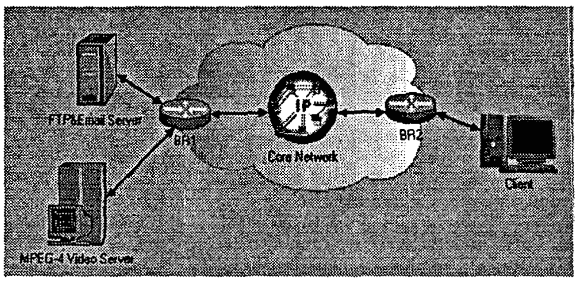

Figure 1. The topology of DiffServ network model

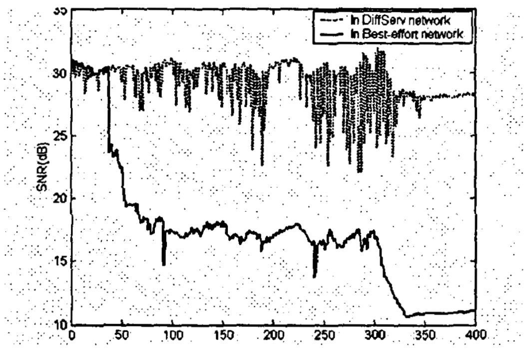

Figure 2. Comparison of PSNR

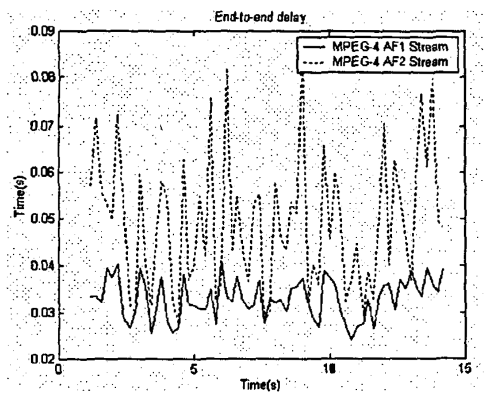

Figure 3. MPEG-4 Packet ETE delay

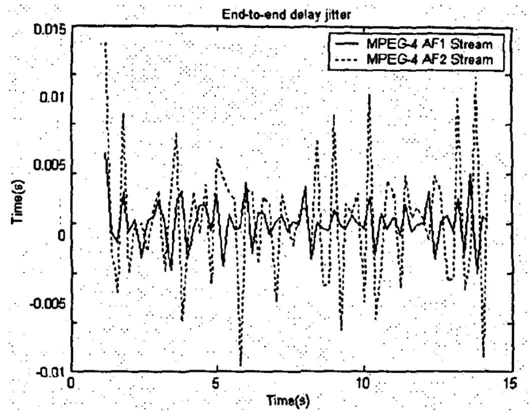

Figure 4. MPEG-4 Packet ETE delay jitter

V. DISSCUSSION AND FUTURE WORK

This paper proposes a wired IP DiffServ network framework suitable for real-time video transmission. It would be interesting to look at the hybrid network case, where some parts of the network are wired and the rest wireless. Here, focus on the wireless Local Area Network (LAN), it might be possible to implement new DiffServ scheduling algorithm cooperating with wireless media access protocol.

REFERENCES

[1] S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang, and W. Weiss, "An architecture for differentiated services,", *RFC 2475, IETF,* Dec. 1998.

[2] V. Jacobson, K. Nichols and K. Poduri, "An Expedited Forwarding PHB," *RFC 2598, IETF,* June 1999.

[3] J. Heinanen, F. Baker, W. Weiss and J. Wroclawski, "Assured Forwarding PHB Group," RFC 2597, IETF, June 1999.

[4] T. Ahmed, G. Buridant, A. Mehaoua, "Encapsulation and Marking of MPEG-4 Video over IP Differentiated Service," in *IEEE ISCC 01.* Tunisia, pp. 346-352, July 2001.

[5] T. Ahmed, A. Mehaoua, G. Buridant, "Implementing MPEG-4 Video On Demand Over IP Differentiated Services," in *IEEE GLOBECOM 2001*, vol. 4, pp. 2489-2493.

[6] P. Cuenca, et al. "Error-Resilient Video Transmission over ATM Networks," *IEEE Communications Magazine*, pp. 106-111, Dec. 1999.

[7] Jun Takahashi, Hideki Tode and Koso Murakami, "QoS Enhancement Methods for MPEG Video Transmission on the Internet," *ISCC'02* , vol.E85-B no.5 pp.1020.

[8] Kwon. Taeck-Geun, Rho. June-Kyung, Lee. Sook-Hyang, "Scheduling algorithm for real-time burst traffic using dynamic weighted round robin," *ISCAS '98*, vol.6, pp. 506 –509.