A Framework for a Distributed Protocol Set to Provide Better Quality of Service for Multimedia Delivery on IP Networks

Hossein Mohammadi
Islamic Azad University of Zanjan
Zanjan, Iran
Hosm@ece.ut.ac.ir

Nasser Yazdani
University of Tehran
Tehran, Iran
Yazdani@ut.ac.ir

Mahmoud R. Hashemi
University of Tehran
Tehran, Iran
Hashemi@comnete.com

Abstract –**Internet is migrating from a simple data network to an environment where a more demanding multimedia content like audio, video, and IP telephony is being delivered. The Internet Protocol (IP) was originally designed to interconnect heterogeneous networks. It scales well by keeping the core network as simple and dumb as possible and provides a best effort delivery service. However, multimedia applications require something better than a simple best effort delivery. Many solutions have been proposed to implement quality of service (QoS) on IP networks. Typically, these methods do not take into account the inherent characteristics of multimedia data, and leave most of the work to the end hosts. In this paper, we propose a framework for a new protocol set to integrate network and application level QoS to reach the best possible quality for multimedia delivery over the Internet. The suggested method has a modular, distributed and scalable architecture, enabling it to easily grow as the network size and/or QoS requirements change.**

Keywords- QoS; Queuing; Classification; Multimedia

I. INTRODUCTION

The initial goal in designing the Internet protocol was connectivity. The main requirement for this network was to provide a survivable connection environment. Initially, the network traffic consisted mostly of simple text data, and quality of service (QoS) wasn't a major requirement. Even today, typical Internet traffic, like web browsing is not sensitive to delay, variations of the delay, and in most cases accidental loss of the packets that are being transmitted, as long as they are kept within a reasonable limit. However real-time multimedia contents such as video are very sensitive to these parameters.

There is no provision within the IP protocol to directly support QoS. Many solutions have been proposed to provide guarantees for the quality of multimedia content delivery. However, they typically provide few classes of quality (e.g.: 64 classes in DiffServ [1]) and they do not support intra class priority. In other word, Network level and Application level Qos are distinct and the network is not aware of the inherent properties of multimedia and non-multimedia traffic.

In this paper, we have introduced a new framework for a distributed protocol set to provide better quality for multimedia applications over the Internet. Our suggested method is simple and can be seamlessly integrated with current network technologies. The proposed protocol is modular, distributed, and highly scalable, enabling it to easily grow as the network size and/or QoS requirements change.

The rest of the paper is organized as follows. In section 2 we review some of the current trends to provide QoS for multimedia content. The proposed protocol is detailed in section 3.Simulation results are presented in section 4. Finally, section 5 concludes the paper.

II. RELATED WORK

Many methods have been proposed to support Qos on IP networks. The major two common methods are Differentiated Service (DiffServ) [1] and Integrated Service (IntServ) [2]. InteServ has many FIFO queues with a specific scheduler that tries to approximate fair queuing. This method can guarantee the service level but because of overhead problems, it can not scale very well. DiffServ scales well by keeping the number of queues as small as 64 and not strictly guaranteeing a service level. However, none of these two schemes does not take advantage of Intra-Queue scheduling. Our approach can integrate with DiffServ and IntServ to aware them of multimedia packets inherit properties. This is done by replacing the queuing system with our PAQ (see section 3).

Some works [3] try to achieve multimedia quality by ignoring the network and supposing it to be like ATM that inherently has Qos mechanisems like similar to IntServ, and then working in the end host. However, these methods are not suitable for IP networks due to their nondeterministic traffic nature.

Queuing mechanisms are also in the field of this paper due to their big impact on delay and jitter. Almost all of the proposed queuing systems try to approximate fair queuing. Some of them like WFQ [4] use weights for queues and some others like CBFQ [5] use crediting to determine each queues output and some of them use priority [6]. All of these methods use multiple queues and their impact is on inter-queue scheduling system. Our PAQ is a stand alone queue not a queuing system, means that one can use some PAQs with a CBFQ scheduler to make a CBFQ that is aware of multimedia properties and tries to minimize multimedia packets delay and jitter.

III. DISTRIBUTED MULTIMEDIA DELIVERY PROTOCOL

Fig.1 illustrates a typical connection between a multimedia server and a multimedia client over an IP network. The system consists of a *Server* providing the content, a *Client* receiving it, *Core Routers* at the center of the IP cloud, and *Edge Routers* at each end. The proposed distributed multimedia delivery protocol (DMDP), has a set of rules for each one of the above components. In this section we introduce DMDP, starting with the initial design goals, and then explaining the protocol architecture.

A. Design Goals

The initial design goals for DMDP were as follows:

1- To deal with the ever increasing traffic of the core network, components are kept stateless and as simple as possible.

2- The protocol must help current QoS technologies to provide a better service quality for multimedia communications while delivering acceptable services for traditional connection oriented communications like TCP.

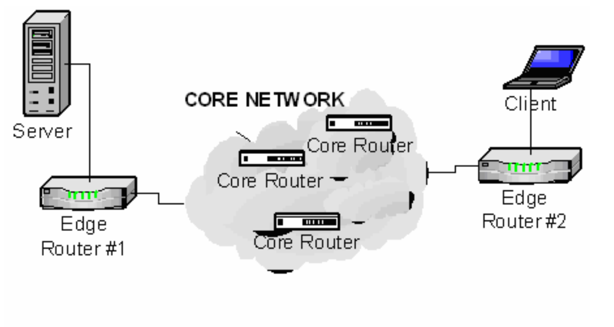

Fig. 1. Components of a typical multimedia delivery on IP Networks

3- The protocol should not put a large overhead on the core network and it should do fate sharing to leave some of load for the end hosts.

4- The protocol should support multiple multimedia classes of service for each host, referred to as *End host service differentiation*.

5- The protocol should be modular, and scalable. The more components we use the better QoS management and flexibility we have. With a modular and scalable architecture one does not need to change all the existing network components in order to benefit from this protocol.

6- The protocol should be able to communicate with current high level and specific multimedia delivery protocols like H.323.

7- The protocol should be able to take advantage of efficient multimedia packet drops to gain more network efficiency. This means dropping a multimedia packet in the edge which we probably know that it will not reach the destination on time.

B. DMDP Architecture

As the name implies, DMDP is a distributed protocol with a set of rules for each one of the components displayed in Fig.1. DMDP easily integrates with existing QoS technologies, including IntrServ [2], DiffServ [1], and best effort, improving the service quality for multimedia content. DMDP consists of the following three components:

i- End Host Component

Relying merely on network level QoS management cannot provide the best service for multimedia content. Simply because at the network level one cannot differentiate the content type, and the special requirements of packets that have the same class of service. This differentiation is implemented at the application level, and performed at the end hosts.

DMDP's architecture at end hosts is illustrated in Fig.2. Solid lines represent data and dotted lines correspond to control. The *Multimedia Server*, an application server delivering multimedia content, separates the content providers including codecs and multimedia devices from the communication components dealing with packet transmission. This approach not only simplifies the codecs and/or applications' structure, but also takes some of the QoS management load off the network.

The *Multimedia Server* also reduces the internal traffic between the multimedia devices and applications or from the applications to the network.

The *QoS Manager* monitors the network traffic, and tries to cope with the variations in available service level caused by changes in network conditions. It does it by controlling the Multimedia Server's output bitrate.

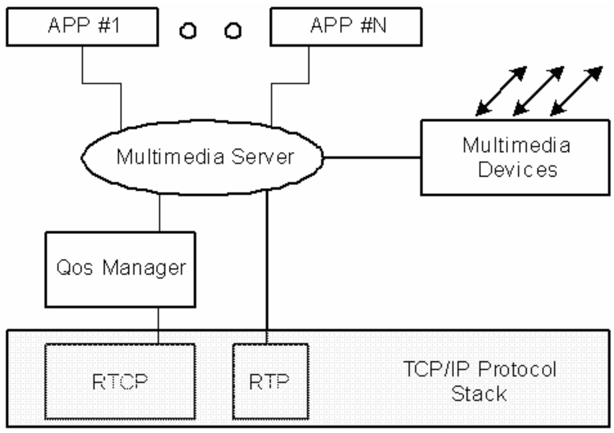

Fig. 2. DMDP at the end host

For any new connection, the client initiates an algorithm similar to the TCP advertised window. This algorithm selects the initial bitrate based on the end host's status, network traffic, and higher level policy protocols (e.g. H.323 gatekeepers). This bitrate is then reported back to the applications through the *Multimedia Server.* The content provider, on the other side follows the same approach.

Thus DMDP defines a framework protocol not specific protocols; both sides can use their own communication protocols in codec and multimedia server level. In other words, having the parts of Fig.2 is optional for each side and they can operate fully independent.

ii- Edge Router Component

The edge router component is illustrated in Fig.3. As a first step, edge routers perform an application level packet classification [7] to filter out multimedia packets from the rest of the traffic. Although this process is normally computationally intensive and inherently slow, but considering the relatively lower traffic of the edge routers, and using fast algorithms such as common multiple-field packet classification algorithms [7], [8] it can be done reasonably well.

In DMDP, the standard router queue has been replaced by a new *Priority Aged Queue (PAQ).* PAQ classifies incoming packets according to their required class of service in corresponding priority queues, and sends them out accordingly. Each packet has two parameters associated with it: Age-Limit, and Priority-Level. The Priority-Level specifies the number of levels a higher priority packet can move ahead in the queue. The Age-Limit specifies the number of time a packet is passed by a higher priority one before it's sent out. The latter parameter is to insure that no low priority packet is left out the queue longer than expected, and that the traditional connection oriented traffics are not affected by DMDP. These parameters are represented by additional bits at the beginning of the RTP packet.

The following pseudo code shows the Enquing method in PAQ. The dequing method PAQ is a simple FIFO queue.

```
method Enque (inputs: Packet P)
Begin
        If PAQ.Empty() OR P.Priority=0 then Insert-at-end (P);
        Else begin
                Packet *tmp = PAQ.tailpacket;
                While (tmp.Age < AgeLimit and P.Priority >
                                tmp.Priority)
                  Begin
                        tmp.Age++;
                        tmp = tmp.getpervious();
                  End while
                PAQ.InsertAfter (tmp,P);
        End else
End Enque
```

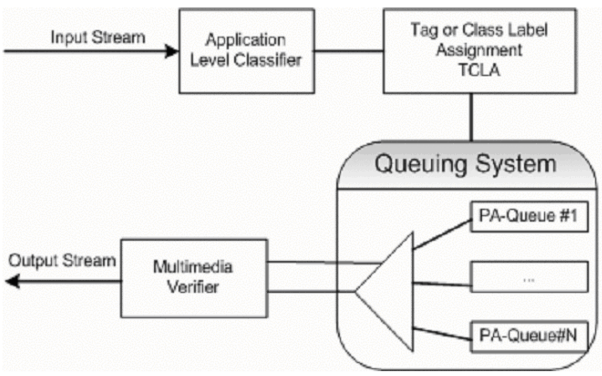

Fig. 3. DMDP at the edge router

Application Level Classifier is a Packet Classifier [7] that assignes a flow id to a packet based on its fields. The assignment is done in the TCLA box. Then, in the queuing system box, a method like WFQ, CBFQ, etc. runs and schedules outputs of the PAQs. At last, Multimedia Verifier verifies the multimedia packets to probably drop or delay them.

The main idea behind this part is to allocate better delay and jitter-services for multimedia packets in each class. For instance, suppose our router is configured to act for three different classes of service and a multimedia packet is sent to second queue, our algorithm tries to 1- reduce its delay by assigning an intra-class priority to it. 2- Sometimes using its jitter tolerance information to select it as a victim to wait. 3- Take advantage of wise drops of multimedia packets to achieve better network utilization. 4- Keeping multimedia connections alive during small congestions.

To achieve these goals, our method assumes RTP as basis for multimedia and assigns some additional intra-class priority bits. These bits are used to identify multimedia packets priority. On the other hand, taking advantage of dropping useless multimedia packets (due to their presentation information) can improve network utilization. To keep multimedia connections alive during small congestion, DMDP gives a second chance to multimedia packets that is selected to be dropped since non-multimedia connections can tolerate jitter and loss better than multimedia. This is because of end-to-end and accumulative nature of TCP [9].

iii- Core Router Component

As mentioned above one of the main design goals in DMDP was keep the core network as simple as possible to insure a quick response time and a scalable architecture. In DMDP we only change the core switch queues to PAQ.

As shown in Fig.4, at the core router packets are classified using the intra-class bits added at the edge router as explained above. This bit classifier can be easily implemented using a simple and fast hardware system..

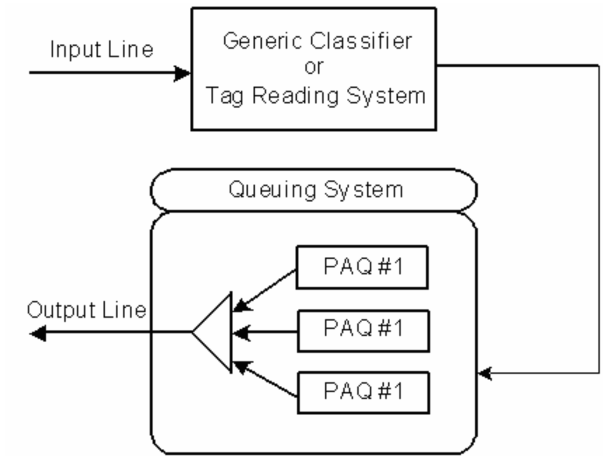

Fig. 4. DMDP at the Core Router

IV. EXPERIMENTAL RESULTS

Priority Aged Queuing was simulated with ns-2 network simulator [10]. In our simulations PAQ was compared with

DropTail (simple FIFO) queue. We have selected FIFO because almost all of the queuing systems use it as their intra queue scheduling but we are using PAQ. For each experiment three different Priority-Level, and Age-Limit values where used. We have selected three important factors of multimedia delivery as comparison-criteria. These factors are: Average bandwidth, Delay and Jitter.

As shown in Fig.5 PAQ provides a higher average bandwidth, an essential requirement for multimedia content. Results indicate that the higher the parameters the higher the bandwidth. Note that as the PAQ parameters increase less bandwidth is allocated for non-multimedia network traffic such as TCP. Hence one should be careful about increasing the parameters too much. Our experiments indicate that values in the range of 10 barely affect the TCP traffic, while still improving the multimedia delivery QoS.

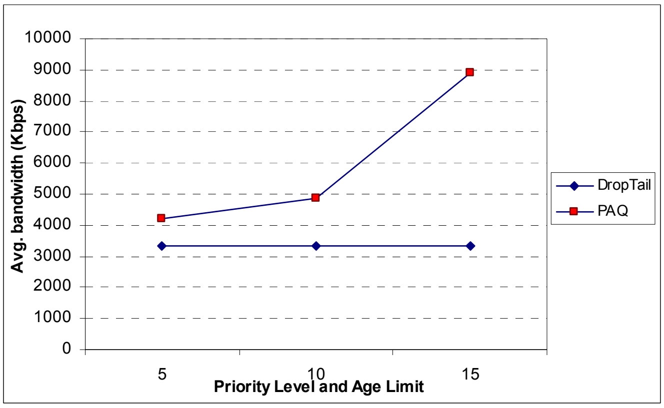

Fig. 5. Average bandwidth using PAQ vs. DropTail.

The average delay is another network feature that affects multimedia content. As illustrated in Fig.6 the average delay for multimedia packets is reduced with PAQ. Results indicate that the delay decreases as the PAQ parameters increase.

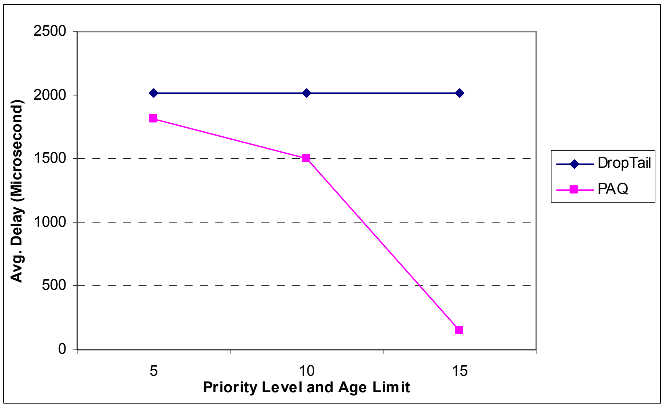

Fig. 6. Average Delays PAQ vs. DropTail.

As shown in Fig.7 PAQ reduces jitter for multimedia content.Results indicate that larger values for PAQ parameters provides a better multimedia quality of service.

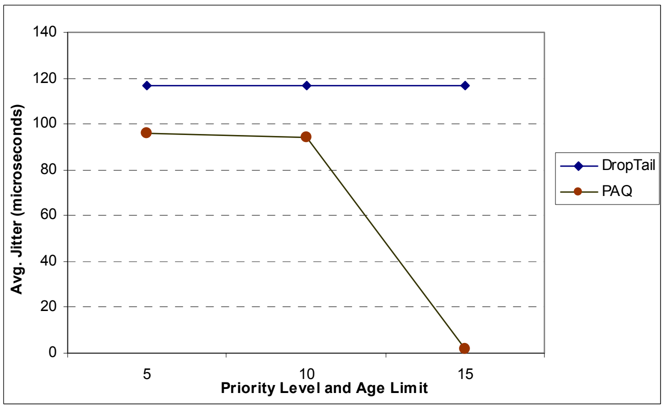

Fig. 7. Average Jitter PAQ vs. DropTail.

V. CONCLUSION AND FUTURE WORK

In this paper we have proposed a new distributed multimedia delivery protocol (DMDP), to provide better quality of service for multimedia content on IP networks. The proposed method is modular, distributed and scalable. In the proposed protocol the core network components are kept as simple as possible to ensure a quick response time and a minimum computational overhead. Simulation results indicate the suggested method provides a better quality of service compared to conventional methods.

The future work will attend to clarify the protocol-set with more details, running large scale simulations and developing a dynamic control algorithm for PAQs parameters.

ACKNOWLEDGEMENT

The authors would like to thank Mr. Sajjad Zarifzadeh for his assistance with the simulations, and Dr. Monadi, Dr. Partovi, Dr. Rahnema and Mr. Teimoorinia from the Islamic Azad University of Zanjan, for their support.

REFERENCES

[1] K. Nichols, S.Blake, F.Baket, and D.L. Black, "Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 headers: RFC 2474 IETF, Dec. 1998

[2] R.Braden, D.Clark, and S.Shenkat, "Integrated Services in the Internet Architecture: An overview," RFC 1633, IETF , June 1994

[3] M. Albrecht, M. Koster, P. Martini, M. Frank, "End-to-end QoS management for delay-sensitive scalable multimedia streams over DiffServ" 25th Annual IEEE Conference on Local Computer Networks (LCN'00) - November 08 - 10, 2000

[4] A. Demets, S.Keshav, and S.Shenkat, "Analysis and Simulation of a Fair Queuing Algorithm," Internetworking: Research and Experience, pp. 3-26, 1990

[5] B. Bensaou, K. T. Chan, and D. H. K. Tsang, "Credit-Based Fair Queueing (CBFQ):A simple and feasible scheduling algorithm for packet networks" Proceedings of IEEE ATM Workshop1997, Spain

[6] H. Khanmirza, S Zarifzadeh, “Cpmparitive Study of ef Behavior in Alternate Priority Queuing” to appear in proceeding of IST 2003 – Isfahan - Iran

[7] Pankaj Gupta, Nick McKeown, "Packet Classification on Multiple Fields", proceeding of ACM SIGCOMM , 1999

[8] Florin Baboescu, George Varghese, "Scalable Packet Classification" Proc. of ACM SigCom 2001, California, USA

[9] RFC 793 “ Transmission Control Protocol” IETF.org

[10] Ns-2 network simulator project, http://www.isi.edu/nsnam/ns/