

Design principles of a QoS-oriented Transport Protocol

Ernesto Exposito¹, Michel Diaz², Patrick Sénac^{2,3}

¹ NICTA, NPC Program, Locked Bag 9013, Alexandria, NSW 1435, Australia
ernesto.exposito@nicta.com.au

² LAAS/CNRS, 7 avenue du Colonel Roche, 31077 Toulouse cedex 04, France
michel.diaz@laas.fr

³ ENSICA, DMI 1 Place Emile Blouin 31056, Toulouse Cedex, France
patrick.senac@ensica.fr

Abstract. In this paper, the needs for specialized end-to-end communication services oriented to satisfy the QoS requirements of current and future multimedia applications are raised. Face to the complexity involved in the wide deployment of QoS guaranteed network services as well as the reduced set of services offered by traditional and recent transport protocols, a QoS-oriented transport protocol (QoSTP) is proposed as the adequate solution for common Internet users for next few years. The design of this QoSTP is based on a set of fundamental principles aimed at assuring the feasibly and efficient deployment of adequate mechanisms regarding the applications requirements. Experimental results demonstrate the feasibility and advantages of this proposal.

1. Introduction

Traditional and new generation of transport protocols have been designed taking into account only a subset of the QoS requirements of multimedia applications. Indeed, these protocols have been mainly focused to the implementation of congestion control mechanisms to save network resources (i.e. TCP, SCTP and DCCP) while providing full order and full reliability or non order and non reliability at all. Moreover, mechanisms intended to satisfy time constraints are not supported at the transport layer. A QoS oriented transport service based on the delay, jitter, throughput and synchronization constrains of multimedia applications and taking into account the partial order and partial reliability tolerance as well as the scalable characteristics of multimedia flows has not yet been provided. In addition, at the network layer, even if a lot of research aimed at the provision of end-to-end QoS guarantees has been carried out, today and for the next few years, the Best-Effort service will be the predominant and more accessible network service in Internet.

These are the reasons that led us to propose the design of a QoS-oriented transport protocol (QoSTP). This design has to be based on a set of fundamental principles aimed at assuring the feasibly and efficient deployment of adequate transport mechanisms regarding the applications requirements.

This paper is organized as follows. Next section introduces the design principles of this QoSTP. Sections 3 and 4 describe respectively the API and the mechanisms of QoSTP. Section 5 presents some experimental results. Finally, the conclusions and perspectives of this work are presented.

2. Design

The design of a QoSTP should answer the following fundamental questions:

- Application Programming Interface (API): how multimedia applications are going to communicate their QoS requirements to the transport protocol, minimizing application adaptation efforts?
- Mechanisms: which transport protocol mechanisms must be deployed to satisfy the application requirements taking into account the available resources and network services?

3. Transport service API

The first design principle of a QoSTP is related to the facilities that have to be offered by its API to the multimedia applications. This API must offer enough expressiveness capabilities in order to allow new multimedia applications to explicitly specify their QoS requirements while preserving its backward compatibility with legacy applications. From the transport protocol point of view, the establishment/termination and transmission phases performed by the multimedia applications are translated in several transport services access points (TSAP): session control and media transmission.

- Session control: Multimedia applications start establishing a session control connection. The messages exchanged over this connection can be very useful for the provision of QoS at the transport layer for the media flows participating in the session. For this reason, a QoSTP should provide session control service (i.e. fully ordered and fully reliable service), but in addition offering the possibility of mapping the application requirements to the adequate transport mechanisms. This connection allows preserving the backward compatibility with legacy multimedia applications which will use this service to establish, control and terminate multimedia sessions. Simultaneously, new multimedia applications could also use this connection to explicitly specify their QoS requirements. The preservation of compatibility with current applications and the offering of new specialized transport services could be assured using standard and opened specification techniques such as the XML-based language specifications [3, 4].
- Media transmission: QoS Control and management mechanisms provided by the QoSTP will operate over the media flows exchanged by applications. These mechanisms require specific QoS information describing the packets composing the media flows (i.e. presentation time, importance degree, dependency, etc). In [2] some considerations for a new generation of protocols based on the Application Level Framing (ALF) concept have been presented. The ALF principle introduces the concept of “transfer syntax” as a mean to specify QoS-related information to every data packet to be transmitted. Currently, a description header preceding the payload of every data packet is a usual way to convey this kind of QoS information between remote applications (i.e. multimedia applications implementing the RTP protocol).

4. Transport mechanisms

Next paragraphs will present a detailed study related to the congestion control and the error control mechanisms intended to deduce the most adequate mechanisms to be provided by a QoSTP.

4.1. Congestion control

The Internet protocol architecture is based on a connectionless end to end packet service using the IP protocol. These characteristics offer advantages in terms of flexibility and robustness, but a careful design is also required to provide good service under heavy load. In fact, lack of attention to the dynamics of packet forwarding can result in severe service degradation. This phenomenon is technically called congestion [8]. Network congestion is characterized by excessive delay and bursts of losses in delivering data packets. Congestion control mechanisms are intended to avoid network congestion and its consequences [7]. Some studies have been done in order to propose congestion control mechanisms adapted to the characteristics of multimedia applications [6]. One of these mechanisms is the TCP Friendly Rate Control or TFRC.

4.1.1. TCP Friendly Rate Control (TFRC)

TFRC is a source and model based congestion control mechanism that provides a TCP friendly sending rate while minimizing abrupt rate changes [9]. TFRC has been designed to be reasonably fair when competing for bandwidth with TCP flows. A flow is considered as being "reasonably fair" if its sending rate is generally within a factor of two of the sending rate of a TCP flow under the same conditions. Furthermore, TFRC has a much lower variation of throughput over time compared with TCP, which makes it more suitable for real-time multimedia applications. The sender sends a stream of data packets to the receiver at some rate. The receiver sends a feedback packet to the sender roughly once every round-trip time (RTT). Based on the information contained in the feedback packets, the sender adjusts its sending rate in accordance with the TCP throughput equation to maintain TCP-friendliness [8].

$$r = s / (R (2p/3)^{(1/2)} + 3RTO (3p/8)^{(1/2)} p (1+32p^2))$$

Where r is the sending rate; s is the mean packet size; RTT is the round trip time; RTO is the retransmission timeout; and p is the loss event rate. If no feedback is received from the receiver in several RTT s, the sender halves its sending rate.

Currently, the TFRC can be considered as the most suitable congestion control mechanism to be implemented by a QoS oriented transport protocol. Indeed, the TCP model based characteristics, the minimization in the abrupt rate changes, as well as the error control independency are some of the required functionalities of a QoSTP. However, this mechanism uses a delaying policy to perform the rate control which could sometimes be non-compliant with delay-constrained multimedia applications (e.g. interactive applications). Next paragraphs introduce an enhancement to the TFRC congestion control mechanism.

4.1.2. QoS-aware TFRC congestion control

The rate control mechanism of TFRC is based on a delaying policy aimed at adapting the flow to the allowed sending rate. This mechanism can penalize applications with strict delay constraints. Indeed, for these applications, received packet could be discarded if they arrive too late to be presented. An alternative to the delaying policy implemented by TFRC may be a quality adaptation policy. Quality adaptation mechanisms can be performed by applications (i.e. adaptive encoding, switching between multiple pre-encoded version, etc.). But usually these mechanisms are executed in long timescales. This proposal consists in performing quality adaptation at the transport level. This requires that QoS information describing the multimedia flows must be available at the transport layer. This information must include at least the time constraints associated to every ADU as well as specific QoS information aimed at

performing the quality adaptation (i.e. ADU priorities, dependency, order, etc.). The delaying strategy of TFRC is based in the computation of the inter-packet interval time (IPIT) for every data packet to be transmitted. TFRC calculates this IPIT value as indicated in (1)

- (1) **IPIT = s/r ;**
- (2) **oneWayDelay = $RTT/2$**
- (3) **eDeliveryTimestamp = now + IPIT + oneWayDelay**
- (4) **eDeliveryTimestamp – timestamp > MAXDELAY**

Where s is the packet size and r is the allowed sending rate. The IPIT value represents the time to be delayed the current data packet in order to respect the allowing sending rate. If the QoS information associated to data packets includes the delivery timestamp of every packet to the receiving application then the feasibility of this delaying strategy could be checked, taking into account the end-to-end delay of the applications. The one-way-delay must be known in order to perform this temporal validation. The one-way-delay can be estimated using the RTT as estimated in TFRC (2).

Using the oneWayDelay value, the delivery timestamp of the current data packet can be calculated as indicated in (3), where now is the current time. Data packet can be considered as obsolete by the receiving application if condition (4) is checked, where timestamp is the scheduled delivery timestamp and MAXDELAY expresses the delay tolerance of the application (i.e. 300 or 400 ms for interactive application).

These obsolete packets will be generally discarded by receiving applications. However, if the temporal validation is performed by the source, discarding could be anticipated in order to avoid the bandwidth being wasted. Nevertheless, this basic discarding policy could seriously affect the QoS perceived by the final user if important ADUs are discarded. For these reason, we propose the use of selective frame discarding methods based on ADU-related QoS information in order to optimize the QoS provided to the user while preserving network resources and respecting the application delay constraints. This selective frame discarding method can be applied if the medium has been encoded using specific compression and ADU segmentation techniques which facilitate the implementation of this method at the transport layer (i.e. ALF approach for the segmentation of flows such as MPEG, H.263, MJPEG, etc.).

The transport level quality adaptation strategy could be defined using the QoS description of the ADUs composing the multimedia flows. This information can be used to define a set of quality layers. For instance, for a MPEG flow composed by I, P and B images, 3 quality layers could be defined:

- Layer 2: I, P and B images**
- Layer 1 only I and P images**
- Layer 0: only I images**

The definition of these different quality layers allow us to propose an enhancement to the TFRC algorithm intended provide a rate control compatible with the time constraints and the intrinsic characteristics of multimedia flows. Next algorithm describes this specialization of the TFRC mechanism:

```

currentLayer=0
join layer(currentLayer)
while (sessionIsActive) {
// When feedback received or noFeedBack timeout : estimate of TFRC parameters & compute r
// filtering
  If (currentPacket.layer>currentLayer) then currentPacket.discard=true
  else
  {
    IPIT = currentPacket.size / r; // inter-packets interval
    eDeliveryTimestamp = (now + t_ipi + RTT/2 + delta) // estimation of time of data delivery
    eDelay = eDeliveryTimestamp - currentPacket.timestamp // est. of presentation delay
    // quality adaptation action in response to the estimated delay
    if (eDelay <= MinDelayThreshold) // i.e. 50 ms
      action=increase
    else (if eDelay >= MaxDelayThreshold) // i.e. 400 ms
      action=decrease
    // quality adaption decrease action
    if (action==decrease)
      if (currentLayer==MIN_LAYER && eDelay <MinDELAY)
        { drop layer(currentLayer)
          STOP
        }
      else if (currentLayer> MIN_LAYER)
        {
          currentLayer=currentLayer-1;
          currentPacket.discard=true
        }
    // quality adaption increase action
    if (action==increase&&currentLayer< MAX_LAYER)
      currentLayer=currentLayer+1;
  }
//scheduling of current packet transmission
if not currentPacket.discard then scheduleTransmission currentPacket,t_ipi
}

```

Where now is the current time, RTT is the round trip time and delta is a tolerance constant including error in time estimations. In order to avoid abrupt changes in the QoS provided to the final user, quality layer increase and decrease actions have been proposed to be tailored by the MinDelayThreshold and MaxDelayThreshold obtained from the QoS requirements.

4.2. Error control

Several multimedia applications present some preference for timeliness over order and reliability [12, 1, 11]. Actually, many of these applications do not require a fully ordered and fully reliable transport service when the delay incurred by this service is not compatible with their time constraints. For this reason, most of multimedia applications have been designed to use the UDP protocol without any guarantees of order and reliability. In some cases, these applications have to implement ad-hoc error control mechanisms to satisfy their requirements. In [5] different error control mechanisms based on the partial ordering and partial reliability constraints of multimedia flows and aimed at improving the QoS delivered to the multimedia applications have been proposed.

4.2.1. Partially Reliable, differentiated and Time-constrained ARQ

ARQ error control mechanisms work as follows: when a loss is detected, the receiver send a feedback message to ask the source to retransmit the message. This means that

a retransmitted packet arrives at least three one-way trip times after the transmission of the original packet. Sometimes, this delay could exceed the delay constraints of the application. However, if the one-way trip time is short, this mechanism could be efficiently used to recover the losses. Time-constrained ARQ mechanisms can be implemented for unicast connections by source-based methods. This error control method is intended to avoid retransmissions of packets that will arrive too late to be presented. The retransmissions can be demanded by the receiver when losses are detected. The source will check the following condition before performing the retransmission:

If (now + RTT/2 < presentationTime) then Retransmission of packet

This mechanism can be easily implemented by the source if QoS information related to the time presentation is available at the transport layer. Indeed this method can be used to provide a differentiated and partially reliable service taking into account the notion of differentiated layers previously introduced. Furthermore, this mechanism could work in combination with the QoS-aware TFRC congestion control and a partially ordered mechanism controlled by the receiver to provide the specialized QoSTP services.

4.3. QoS transport mechanisms

This section has presented a proposal including a set of transport mechanisms to be implemented by a QoSTP (see figure 1).

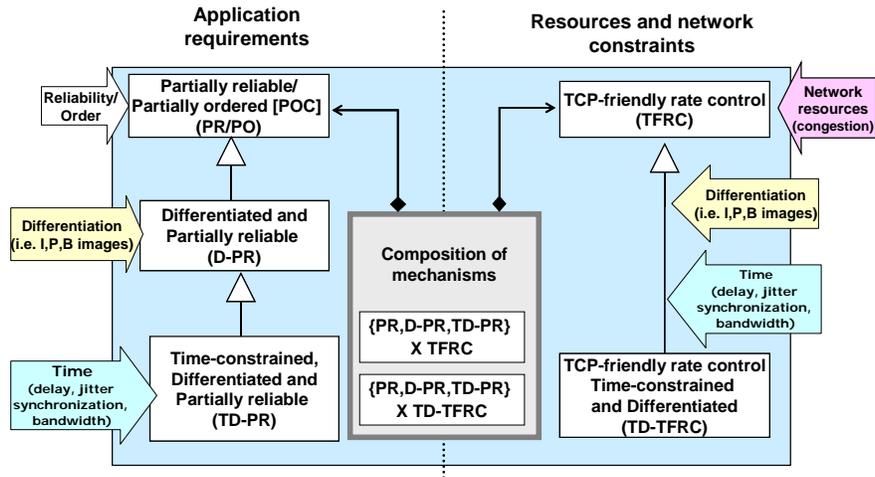


Figure 1: Transport mechanisms to be provided by the QoSTP

These mechanisms include a congestion control mechanism intended to preserve the resources of networks providing a Best-Effort service. An error control mechanism intended to provide a partially ordered and partially reliable service explicitly or implicitly configured from the application requirements has also been presented. This error control mechanism has also been enhanced in order to provide a differentiated and partially reliable service. Both error and congestion control mechanisms have been enhanced to take into account intrinsic application time constraints. Furthermore, the composition of error and congestion control mechanisms provides a large set of transport services for different multimedia applications. Next section presents some experiments intended to evaluate the QoSTP services.

6. Experiments

In [5] several experiments involving interactive and Video on Demand (VoD) applications have been carried out to validate different transport service specifications and to evaluate the protocol performance. In this paper, the experiments intended to evaluate the congestion control mechanisms will be shown. These experiments have been carried out using streaming applications with specific time constraints and using a Java-based QoSSTP implementation following the UML 2.0 specification presented in [5]. In these experiments, Best Effort network services have been emulated using the Dummynet emulator [10]. The emulation scenarios have been characterized by one-way delay of 50 ms and specific bandwidth limitations. This emulation based on bandwidth limitation is intended to create temporal network congestion (i.e. when application sending rate is higher than limited bandwidth).

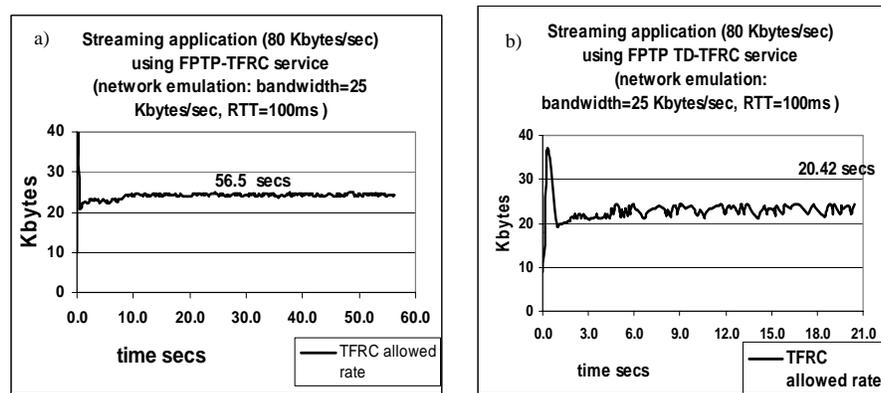


Figure 2: TFRC vs TD-TFRC mechanisms with bandwidth limited to 25 Kbytes/sec

A first experiment involving a streaming application producing a multimedia flow of 80 Kbytes/sec (i.e. H.263 video flow composed by I and P pictures) during 20 seconds and with network bandwidth limited to 25 Kbytes/sec has been carried out. The conformance of the standard TFRC congestion control mechanisms implemented by QoSSTP has been validated (i.e. sending rate according TFRC specification). However, the time required to complete the transmission under this emulated network scenario was of 56.5 seconds with an accumulated delay of 36.5 secs. (see figure 2.a). This delay is not admissible either for interactive or VoD time-constrained applications. In contrast, the time constrained and differentiated specialization of this mechanism (TD-TFRC) mechanism was able to adapt the video flow using the selective discarding policy (i.e. by taking advantage of partial reliability tolerance) in order to respect time constraints as shown in figure 2.b. These results show the advantages of providing adequate congestion control mechanisms for the transmission of scalable media flows with specific time constraints.

Conclusions and perspectives

In this paper the main principles aimed at designing a QoSSTP have been presented. The API of the QoSSTP has been defined as an enhancement of standard BSD socket interface including session and media control connections. On one hand, the session control connection allows preserving the backward compatibility with legacy multimedia applications which will use this service to establish, control and terminate their multimedia sessions. Moreover, new multimedia applications could also use this connection to explicitly specify their QoS requirements related to the multimedia flows. On the other hand, media connections are intended to provide QoS Control and management mechanisms to operate over the media flows exchanged by applications. The transport mechanisms to be implemented by a QoSSTP have also been proposed. These mechanisms include standard congestion and error control mechanisms. Both mechanisms have been enhanced to take into account intrinsic media flow requirements and time constraints. Experimental results have demonstrated that a QoS oriented transport protocol implemented over a Best Effort network service can largely improve the QoS offered to multimedia applications. Future works to develop adequate mechanisms intended to take into account specialized communication services such as guaranteed network services should be done. Likewise, the specialization of current transport mechanisms in order to take into account specific network characteristics such as mobility, wireless and satellite networks will be carried out.

References

1. Amer, P., Chassot, C., Connolly, C., Conrad, P., Diaz, M.: Partial Order Transport Service for MM and other Appli. *IEEE/ACM Trans. on Networking*, vol.2, n5 (1994)
2. D. Clark and D. Tennenhouse, "Architectural considerations for a new generation of protocols", *IEEE SIGCOMM'90*, Sept. 1990
3. E. Exposito, M. Gineste, R. Peyrichou, P. Sénac, M. Diaz, S. Fdida, "XML QoS specification language for enhancing communication services", 15th International Conference on Computer Communication (ICCC'2002), Mumbai (India), August 2002
4. E. Exposito, M. Gineste, R. Peyrichou, P. Sénac, Michel Diaz, "XQOS: XML-based QoS specification language", MMM'03 The 9th International Conference on Multi-Media Modeling, January 7-10, 2003, Taiwan.
5. E. Exposito, "Specification and implementation of a QoS oriented transport protocol for multimedia applications", PhD dissertation, Institut National Polytechnique de Toulouse. December 2003, Toulouse, France.
6. S. Floyd and K. Fall, "Promoting the use of end-to-end congestion control in the Internet," *IEEE/ACM Transactions on Networking*, vol. 6, Aug. 1999
7. V. Jacobson, "Congestion avoidance and control", *ACM SIGCOMM'88*, pp. 314-329, Aug. 1988.
8. Floyd S., "Congestion Control Principles", IETF Request for Comments 2914, September 2000
9. Handley M. et al, " TCP Friendly Rate Control (TFRC): Protocol Specification" , IETF RFC 3448, 2003
10. Rizzo L., "Dummysnet: a simple approach to the evaluation of network protocols", *ACM Computer Communication Review*, Vol. 27, no. 1, January 1997
11. Luis Rojas-Cardenas, Emmanuel Chaput, Laurent Dairaine, Patrick Sénac, Michel Diaz, Video Transport Over Partial Order Connections, in *Computer Networks and ISDN Systems*, vol. 31, Issue 7, Elsevier, April, 1999, pp. 709-725
12. Sénac P., Exposito E., Diaz M., "Towards a New Generation of Generic Transport Protocols", *Lecture Notes in Computer Science 2170*, Springer, Eds. S.Palazzo, September 2001