Abstract

The Internet traditionally provides service that is commonly characterised as a best-effort service. Many applications run very well using this service model but some new interactive applications such as telephony or video conferencing impose stringent demands to the network. In this paper we discuss a proposal for changing the architecture of IP networks in order to provide some level of service discrimination. Architecture for differentiated services (DS) and proposed mechanisms have been analysed and discussed to great extent in the past years. The two most known approaches are schemes for throughput sensitive traffic and schemes for delay sensitive traffic. Differentiated services are attractive because of maintaining connectionless data transfer in IP networks without the need for per-flow state information in network elements and because of using simple signalling. In the paper we outline both principles for service discrimination and then we use simulation to evaluate them. Scheme for the delay sensitive traffic is realised by strict priority queuing in network elements, while scheme for throughput sensitive traffic is realised through pre-emptive buffer processing policy. With simulations we evaluate behaviour of a differentiated network architecture for two widely used (WWW, FTP) and two emerging Internet applications (video conferencing and IP telephony) and compare them with the best-effort network. The results show that proposed schemes have great positive impact on real-time applications that are assigned “high priority”. Queuing delays are reduced to great extent and packet delivery without losses in queues is possible. On the other side, performance of elastic applications is not highly degraded.

1. Introduction

Currently Internet offers only a single best-effort service class. This architecture has been tremendously successful in supporting elastic data applications, like www, e-mail, ftp and news, but it is not able to satisfy demands of new and emerging real-time data applications like IP telephony and video conferencing. Real-time applications have stringent demands to the transmission network and one of the most challenging is delivery of real-time data packets in time for reconstruction of original signal at the receiving end. In other words, real-time data packets must be delivered within certain, mostly very tight, delay bounds.

The Internet research community has devoted much effort to designing architecture capable of supporting real-time applications. The two known principles are integrated and differentiated services. Also strong economic interest exists to integrate business critical, real-time and other data applications onto a single corporate network. Therefore, we must analyse the performances of typical network applications and understand their behaviour in particular network architecture.

In this article we simulate two mechanisms for differentiated services architecture: priority queuing and priority queuing with pre-emption policy. We observe performances of four common applications: www, ftp, real-time video and real-time audio.

2. Properties of network applications

The efficiency of network architecture should not be evaluated merely by technical measures like bandwidth, packet loss level and delay but also by effectiveness of satisfying user needs.

Different applications have different sensitivity to available bandwidth, delay or packet loss level. Based on sensitivity to delay, applications can be divided into two categories: elastic and inelastic (rigid). The performance of elastic applications is not greatly dependent on the delay of individual packets. Examples of such applications are WWW, FTP and electronic mail. On the other hand, performance of inelastic applications is greatly affected by the delay of individual packets. They are not able to adapt to a wide range of a packet delay and delay variance (jitter) at the transmission over data networks. The most characteristic inelastic application is transmission of voice over data networks (IP telephony).

At present economic reasons are the main driver for the introduction and deployment of real-time voice transmission over data networks (particularly those based on Internet Protocol) that were basically designed for the transmission of elastic traffic. That observation is true for both public and private IP networks. Due to a high level of static resource sharing (at all times resources are available to all users that access the network), the network can not offer any guaranties for packet delivery or their delay.
A very important factor for the performance of interactive real-time applications is the delay of reconstructed signal because with the increase of delay the communication becomes first disturbed and later impossible. Multimedia applications, for instance, acquire the signal at the source, perform its coding and packetization and send it to its destination. Data networks inevitably introduce packed delay that is variable in most networks. Variation in delay is called jitter. At the other end the receiver tries to reconstruct the original signal from the arriving packets. To compensate for the variable packed delay as seen at the receiving end the playback of the signal is delayed for a fixed amount of time regarding to packet creation time. Receiver maintains a buffer that stores the packets waiting for the playback. Every packet that arrives before its scheduled playback time can be used in reconstructed signal, packets that arrive after their playback time can not be used and are therefor discarded by the receiver.

Transmission of speech is very sensitive to delay and according to ITU-T recommendation G.114 the one-way delay on the entire transmission path (from acquisition to playback) should not exceed 150ms \[5\]. The rigidity of real-time voice application on data networks does not apply to a certain absolute delay value above which the quality of reconstructed signal would fall drastically, but to the fact that packets that arrive too late can not be used in reconstructed signal at the receiving end.

3. Traffic differentiation

Delays that occur at the transmission of packets over the network consist of a fixed and of a variable part. The fixed part of the delay does not depend on network load but only on link speeds, capacity of network elements and distance between origin and destination. The variable part of the delay is the result of variable queue lengths in network elements that depend mainly on the network load.

Traffic differentiation does not affect fixed part of the transmission delay but can effectively reduce its variable part. In packet networks all packets that can not be serviced (forwarded) immediately are placed into input or output queues in network elements. The next packet from the queue to be serviced is chosen according to a service discipline of that network element. In Internet the service discipline is FIFO (First In First Out) that means that packets are serviced exclusively according to their arrival time at the particular network element.

But not all the packets in the queue are of the same importance or require the same service. Traffic differentiation based on delay requirements exploits the ability of network elements to service packets according to their priority information that is included in their headers: priority field, IP address, protocol, etc. Packets with higher priority are serviced before packets with lower priority. This service discipline is called strict priority queuing.

Different applications have different sensitivity to packet loss. At network congestion times queues in network elements fill-up and arriving packets that can not be placed into the queue are discarded. Probability of packet loss because of full queues can be influenced with traffic differentiation based on throughput. Network elements take priority of packets into consideration and allow packets with higher priority to supplant packets with lower priority. This service discipline is called strict priority queuing with pre-emption.

In the following paragraphs we present simulation results for four typical Internet applications using three different service disciplines in network elements.

4. Network layout and parameters

To determine the influence of a service discipline on the performance of four typical IP applications we have set up a simple network layout with four traffic sources (one for each application) and two network nodes (routers). The layout is presented in Figure 1.

![Figure 1: Simulation network layout](image-url)

Direction of traffic flow was from node 1 to node 2. In the opposite direction only confirmation packets were carried. The link speed was 2048 Mbit/s, full duplex, and it had 10 ms propagation delay. Each node had two input and two output queues, one for each direction. The length of output queues was limited to 5120 bytes.

For each output queue we have set three different service disciplines in three different simulation scenarios: FIFO, strict priority queuing and strict priority queuing with pre-emption.

Applications were modelled by appropriate distribution of interpacket times according to their behaviour and typical traffic parameters. Basic traffic parameters are listed in Table 1. FTP source has generated files of size 100 Kbytes that were partitioned into packets.
largest allowed packet on the network was limited to 1500 bytes.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Speech</th>
<th>Video</th>
<th>WWW</th>
</tr>
</thead>
<tbody>
<tr>
<td>min packet length</td>
<td>30 byte</td>
<td>640 byte</td>
<td>10 byte</td>
</tr>
<tr>
<td>max packet length</td>
<td>30 byte</td>
<td>640 byte</td>
<td>1400 byte</td>
</tr>
<tr>
<td>packet length</td>
<td>constant</td>
<td>uniform</td>
<td></td>
</tr>
<tr>
<td>message length</td>
<td>constant</td>
<td>exponent</td>
<td></td>
</tr>
<tr>
<td>protocol stack</td>
<td>RTP/UDP/IP</td>
<td>TCP/IP</td>
<td></td>
</tr>
</tbody>
</table>

Table 1: Traffic parameters

Applications were using two different protocol stacks: RTP/UDP/IP and TCP/IP that best suited their needs. The former was used for the transmission of speech and video packets and the later for WWW and FTP. TCP sliding window was set to 16 packets. Priorities were assigned as follows:

- Priority 1 FTP (lowest)
- Priority 2 WWW
- Priority 3 Video
- Priority 4 Speech (highest)

For pre-emption schemes only priority 3 and 4 could pre-empt (supplant) packets of lower priorities. That means that speech packets could pre-empt all other packets and video packets only WWW and FTP packets. WWW could not pre-empt FTP packets.

5. Simulation parameters

Each simulation is a combination of three simulation variables:

- a share of real-time traffic,
- service discipline,
- link utilisation.

The share of real-time traffic ranges between 10% and 90%. It represents the share of real-time traffic in comparison to the total traffic on the link and not the share of the real-time traffic comparing to the link capacity. The three service disciplines simulated were FIFO or best-effort (BE), strict priority queuing or differentiated services (DS) and strict priority queuing with pre-emption (DSpre). We have chosen the amount of the entire traffic generated (real-time and non-realtime) in a way to achieve 50%, 90% and 100% link utilisation.

In the following text abbreviations for each simulation scenario (combination of simulation variables) will be used. For instance, DS50 means that strict priority queuing was used and the link utilisation was 50%. The share of real-time traffic ranged between 10% and 90% for each of the combinations of service disciplines and link utilisation.

6. Simulation results

The motivation for a detailed study of delays, throughput and packet loss in IP networks with different service disciplines comes from expected advantageous results that can be achieved. Similar simulations have been presented in [1]. Current simulations are more accurate and closer to the real networks because more network parameters were modelled or taken into consideration (TCP window length, limited buffer space, propagation delay, etc). The results of simulations are described and graphically represented below.
The first set of simulations is the current state of the Internet and serves as the reference for the other two sets. Scenarios BE50, BE90, BE100 (congestion) shown in Figure 2 to Figure 4 demonstrate the behaviour of IP network under different link utilisation.

Delays are equally distributed between all traffic types (applications) with slightly higher delays for FTP traffic because of its on average longer packets. The effect of longer packets is specially expressed in BE50 scenario (Figure 2). The lower delay limit for all traffic is 10ms (propagation delay) and the rest is mostly queuing delay in network nodes except for FTP traffic where link transmission time\(^1\) plays an important role (because of long packets).

With the introduction of priority queuing (Figure 5 to Figure 7) into IP networks, things change quite dramatically. At 50% link utilisation there is virtually no difference. That was expected because on a half empty network FIFO queuing does just fine. But the closer we are to congestion the bigger the differences are.

The first application to suffer is FTP because of its lowest priority. Even at 90% link utilisation the FTP delays rise considerably and reach high values at 100% link utilisation. WWW traffic experience slight delay increase at 90% and suffers high delays only at 100% link utilisation and a high share of real-time traffic. If we compare BE and DS scenarios for FTP traffic we can say that it is certainly on the loosing side. With the introduction of priority queuing FTP shows higher delays at all simulation parameter combinations. That was of course expected because of its lowest priority. WWW traffic has somehow retained almost a “status quo”. It suffers higher delays in DS comparing to BE scenarios only at 100% link utilisation and a high share of real-time traffic.

The winners here are clearly voice and video traffic. With the introduction of DS their delays are almost unchanged regardless of the link utilisation or the share of real-time traffic. The benefits are the most noticeable at 100% link utilisation. Voice traffic gains about 8ms or 40% in delay value (12ms at DS100 comparing to 20ms at BE100). The benefits go on the account of FTP and WWW traffic that experience higher delays.

The last set of simulations are scenarios with strict priority queuing with pre-emption (DS50pre, DS90pre and DS100pre). This service discipline allows high priority packets to pre-empt lower priority packets. That means that if an input or output queue of a network element (node) is full a low priority packet(s) is removed from that queue to make place for a higher priority packet.

Simulation results for DSpre scenarios are presented in Figure 8 to Figure 10. As can be seen they are very similar to simulation results of DS scenarios (Figure 5 to Figure 7), showing that pre-emption does not significantly improve average delay characteristics of high priority packets while low priority packets also do not suffer any significant delay deterioration.

While pre-emption does not greatly affect average packed delay it does have effect on packet loss. Figure 11 to Figure 13 show packet loss results for BE100, DS100 and DS100pre scenarios. When those results are compared it can be seen that there is no significant difference between BE100 and DS100 scenarios, both have approximately the same percent of lost packets.

\(^1\) Time to put the packet on a link
because neither of service disciplines (best-effort and strict priority queuing) can affect placement of packets in the queue at their arrival at the network element or have the ability to discard packets from it when some higher priority packets arrive. But with the introduction of strict priority queuing with pre-emption (DS100pre) loss for voice and video traffic that have high priority falls to 0% because they can always find a place in the queue. This have consequences for WWW and FTP traffic that suffer greater loss because packets are not lost only because of full queues but some packets that are already in the queue are now ejected because of the pre-emption policy.

We have seen that high priority voice and video traffic benefit from traffic differentiation. But that is not the whole story, what about the disadvantages for low priority WWW and especially FTP traffic?

With Figure 14 and Figure 15 we try to show that FTP traffic with the introduction of differentiation does not suffer serious performance degradation. That is a very important conclusion pointing out that there are more benefits than drawbacks with introduction of traffic differentiation.

Figure 14 shows the throughput ratio between DS and BE scenarios at 90% and 100% link utilisation. As can be seen throughput ratio for 90% link utilisation is very...
close to 1 for any share of real time traffic. That means that FTP throughput have not suffered with the introduction of differentiation only its average packet delay has increased (compare Figure 3 and Figure 6). The situation is a little bit different at 100% link utilisation (congestion) where FTP throughput ratio starts to fall above 60% share of real-time traffic because of its lowest priority.

Figure 14: Throughput ratio for FTP traffic

Figure 15 shows and compares the average FTP transfer times for 100kB files. Transfer times for BE and DS scenarios do not differ much, except at 100% link utilisation and real-time traffic share over 60% (because of low throughput). From this result it can be presumed that average packet delay does not have great effect on the performance of elastic applications. For the confirmation of this presumption more simulations with a broader parameter list and measurements on real systems should be done.

Figure 15: Transfer times for FTP files of size 100 KB

7. Conclusion

With real-time IP applications and penetrating business and commercial use it is evident that some sort of service differentiation will have to be introduced into Internet. One of the most critical issues is delivery of packets inside certain delay bounds. Generally speaking IP networks need to be QoS (Quality of Service) enabled.

Much can be done only with changing the queue service discipline inside existing network elements that have that possibility (and most of them already do). Performance of real-time applications (voice, video) improves considerably and low priority applications performance (WWW, FTP) does not suffer too much. The last result should be proved with more extensive simulations and measurement on real systems.

8. Literature


Curriculum Vitae

Anton Kos was born in Jesenice, Slovenia, on June 15, 1969. He received the B.E. degree in 1994 and the M.S. degree in 1998 both from University of Ljubljana, Faculty of Electrical Engineering. He is currently a researcher at the Faculty of Electrical Engineering, Ljubljana. He is a member of the Laboratory for Communication Devices where he is involved in the research of real-time data transmission over packet networks with emphasis on IP networks.

Borut Klepec was born in Murska Sobota, Slovenia on October 10, 1966. He received the B.E. degree in 1991 and the M.S. degree in 2000, both from University of Maribor, Faculty of electrical engineering and computer science. He is currently an IT systems program specialist at the Triglav Insurance Company. His research interest includes quality of service, real-time applications and IP networks.