UDC 621.321.1.519.872

QUALITY OF SERVICE AT THE COMPUTER NETWORKS BASED ON INTERNET

N.I. LISTOPAD, A.G. KOPACHEV, A.A. MATRUK

A framework for the emerging computer networks quality of Internet service is presented. Two important components of the framework are considered: integrated service and differentiated service. They are described and problems related to their implementation are discussed. Two broadly classified algorithms are investigated: scheduling and queue management algorithms. The model of the average time delay τ_i for the type i packet is presented.

Today's Internet only provides best effort services when traffic is processed as quickly as possible, but there is no any guarantee as to timeliness or actual delivery, with the rapid transformation of the Internet into a commercial infrastructure, demands for service quality have developed [1−3].

It is becoming apparent that several service classes are much demanded. One of that service class will provide predictable Internet services for companies that do business on the web. These companies are going to pay a certain price to make their services reliable and to give their users high speed access for their web sites. Another service class will provide low delay and low jitter services to applications such as Internet telephony and video conferencing. Companies are going to pay a premium price to run a high quality videoconference to save travel time and other costs. Finally, the best effort service will remain for those customers who need only connectivity.

Which mechanism should be chosen to provide QOS is a very topical issue. There is the opinion which asserts that *fibers and wavelength division multiplexing* (WDM) will make bandwidth so abundant and cheap that QOS will be automatically delivered. The other opinion asserts that no matter how much bandwidth the network can provide, new application will be invented to consume them.

Therefore mechanisms will still be needed to provide QOS. Here you should simply note that, even if bandwidth will eventually is not going to happen soon, for now, some simple mechanisms are definitely needed in order to provide QOS on the Internet [2].

The *Internet engineering task force* (IETF) has proposed many service models and mechanisms to meet the demand for QOS. Among them are the integrated services/RSVP model, the differentiated services (DS) model, traffic engineering and constraint based routing [1,3].

The integrated service model is characterized by resource reservation for real time applications, before data are transmitted. The applications must first set up paths and reserve resources. So, RSVP is a signaling protocol for setting up paths and reserving resources.

The integrated service model proposes two service classes in addition to best effort service [2, 3]. They are:

1) *guaranteed service:* for applications requiring fixed delay bound;

2) *controlled load service:* for applications requiring reliable and enhanced best effort service.

The paradigm of this model is that «there is an inescapable requirement for routers to be able to reserve resources in order to provide special QOS for specific user packet streams, or flows. This in turn requires flow-specific state in the routers».

As mentioned above RSVP was developed as a signaling protocol for applications to reserve resources. The signaling process is illustrated in fig. 1: the sender sends a PATH message to the receiver specifying the characteristics of the traffic. Every intermediate router along the path forwards the PATH message to the next hop determined by the routing protocol. Upon receiving a PATH message, the receiver responds with a RESV message to request resources for the flow. Every intermediate router along the path can reject or accept the request of the RESV message. If the request is rejected, the router will send an error message to the receiver, and the signaling process will terminate. If the request is accepted, link bandwidth and buffer space are allocated for the flow and the related flow state information will be installed in the router.

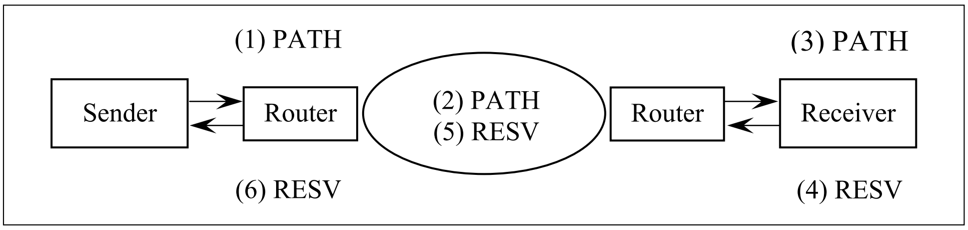

Fig. 1

Integrated services are implemented by four components: the signaling protocol (e.g. RSVP), the admission control routine, the classifier and the packet scheduler. Applications requiring guaranteed service or control- load service must set up the paths and reserve resources before transmitting their data. The admission control routines, will decide whether a request for resources can be granted. When a router receives a packet, the classifier will perform a *multi-field (MF)* classification and put the packet in a specific queue based on the classification result. The packet scheduler then schedules the packet according to QOS requirements.

The integrated services/RSVP architecture represents a fundamental change to the current internet architecture, which is based on the concept that all flow-related state information should be in the end systems.

The problems with the integrated services architecture are the following [3].

1. The amount of state information increases proportionally with the number of flows. This places a huge storage and processing overhead on the routers, therefore, this architecture does not scale well in the Internet core.

2. The requirements on the router are high. All routers must implement RSVP, admission control, MF classification and packet scheduling.

3. Ubiquitous deployment is required for guaranteed service. Incremental deployment of controlled-load is possible by deploying controlled-load service

and RSVP functionality at the bottleneck nodes of a domain and tunneling the RSVP messages over other part of the domain.

Due to the difficulty in implementing and deploying integrated services and RSVP, differentiated services (DS) is introduced. IPv4 header contains a TOS (type of service) byte. Applications can set three bits in the TOS byte to indicate the need for low delay or low loss rate service. However, choices are limited. Differentiated service defines the layout of the TOS byte (termed DS field) and a base set of packet forwarding treatments (termed per-hop behaviors, or PHBs). By marking the DS fields of packets differently, and handling packets based on their DS fields, several differentiated service classes can be created. Therefore, differentiated services form a relative-priority scheme.

Customers can mark DS fields of individual packets to indicate the desired service or have them marked by the leaf router based on MF classifications at the ingress of the ISP (Internet service provider) networks; packets are classified, policed and possibly shaped. The classification, policing and shaping rules used at the ingress routers are driven from the *SLAs* (service level agreements). The amount of buffering space needed for these operations is also driven from the *SLAs*. When the packet inters one domain from another domain, its DS field may be re-marked as determined by the SLA between the two domains.

Differentiated service is different from integrated service. First, there are only a limited number of service classes indicated by the DS field. Since service is allocated in granularity of a class, the amount of state information is proportional to the number of classes rather than the number of flows. Differentiated service is therefore more scalable. Second, sophisticated classification, marking, policing and shaping operations are only needed at boundary of the networks. ISP core routers need only to implement behavior aggregate (BA) classification. Therefore, it is easier to implement and deploy differentiated services.

There is another reason why the second feature is desirable for ISPs. ISP networks usually consists of boundary routers connected to customers and core routers/switches interconnecting the boundary routers. Core routers must forward packets very fast and therefore must be simple. Boundary routers need not forward packets very fast because customer links are relatively slow. Therefore, they can spend more time on sophisticated classification, policing and shaping. Boundary routers at the network access points (NAPs) are exceptions. They must forward packets very fast and so sophisticated classification, policing and shaping. Therefore, they must be well equipped.

In the differentiated services model incremental deployment is possible for assured service. DS-incapable routers simply ignore the DS fields of the packets and give the assured service packets best effort service. Since assured service packets are less likely to be dropped by DS-capable routers, the overall performance of assured service traffic will be better than the best effort traffic.

In this paper a service architecture for differentiated service is presented. The architecture provides assured service, premium service in addition to best effort service.

Assured service is intended for customers that need reliable services for their service providers, even in time of network congestion. Customers will have *SLAs* with their ISPs. The *SLAs* will specify the amount of bandwidth allocated for the customers. Customers are responsible for deciding how their applications share that amount of bandwidth. *SLAs* for assured service are usually static, meaning that the customers can start data transmission whenever they want without signaling their ISPs.

At present there are two directions of investigating broadly classified under the monikers «scheduling algorithms» and «queue management algorithms» [2]. The generic scheduling algorithm, exemplified by the well-known Fair Queuing (FQ) algorithm, requires the buffer at each output of a router to be partitioned into separate queues each of which will buffer the packets of one of the flows. Packets from the flow buffers are placed on the outgoing line by a scheduler using an approximate bit-by-bit, round-robin discipline. Because of per flow queuing, packets belonging to different flows are essentially isolated from each other and one flow cannot degrade the quality of another. However, it is well-known that this approach requires complicated per flow state information, making it too expensive to be widely deployed.

To reduce the cost of maintaining flow state information, I.Stoica has recently proposed a scheduling algorithm called Core Stateless Fair Queuing (CSFQ) [4]. In this method routers are divided into two categories: edge routers and core routers. An edge router keeps per flow state information and estimates each flow's arrival rate. These estimates are inserted into the packet headers and passed on to the core routers. A core router simply maintains a stateless FIFO queue and, during periods of congestion, drops a packet randomly based on the rate estimates. This scheme reduces the core router's design complexity. However, the edge router's design is still complicated. Also, because of the rate information in the header, the core routers have to extract packet information differently from traditional routers.

Another notable scheme which aims to approximate FQ at a smaller implementation cost is Stochastic Fair Queuing (SFQ) proposed by McKenny [5]. SFQ classifies packets into a smaller number of queues than FQ using a hash function. Although this reduces FQ's design complexity, SFQ still requires around 1000 to 2000 queues in a typical router to approach FQ's performance.

Other directions of algorithms, queue management algorithms have had a simple design from the outset. Given their simplicity, the hope is to approximate fairness. This class of algorithms is exemplified by Random Early Detection (RED) proposed by S. Floyd and V. Jacobsen [5, 6]. A router implementing RED maintains a single FIFO to be shared by all the flows, and drops an arriving packet at random during periods of congestion. The drop probability increases with the level of congestion. Since RED acts in anticipation of congestion, it does not suffer from the «lock out» and «full queue» problems inherent in the widely deployed Drop Tail mechanism. By keeping the average queue-size small, RED reduces the delays experienced by most flows. However, like Drop Tail, RED is unable to penalize unresponsive flows, which are based on UDP datagram protocol. This is because the percentage of packets dropped from each flow over a period of time is almost the same. Consequently, misbehaving traffic can take up a large percentage of the link bandwidth and starve out TCP friendly flows and by this it does harm to the end users.

Premium service provides low-delay and low jitter service for customers that generate fixed peak bit- rate traffic. Each customer will have a SLA with its ISP. The SLA specifies a desired peak bit- rate for a specific flow or an aggregation of flows. The customer is responsible for not exceeding the peak rate. Otherwise, excess traffic will be dropped. The ISP guarantees that the contracted bandwidth will be available when traffic is sent. Premium service is suitable for Internet telephony, video conferencing, or for creating virtual lease lines for private networks (VPNs).

Because premium service is more expensive than assured service, it is desirable for ISPs to support both static *SLAs* and dynamic *SLAs*. Dynamic *SLAs* allow customers to request for premium service on demand without subscribing to it. Admission control is needed for dynamic *SLAs*.

Premium service can be implemented as follows. At the customer side, some entity decides which application flow can use premium service. The leaf routers directly connected to the senders will do MF classifications and shape the traffic. Conceptually, we can consider that there is a P-bit in the DS field. If the P-bit of a packet is set, this packet belongs to the premium class. Otherwise, the packet belongs to the assured service class or best effort class. After the shaping, the P-bits of all packets are set for the flow that is allowed to use premium service. The exit routers of the customer domain may need to reshape the traffic to make sure that the traffic does not exceed the peak rate specified by the SLA. At the provider side, the ingress routers will police the traffic. Excess traffic is dropped and all packets with the P-bit set enter a premium queue (PQ). Packet in the PQ will be sent before packets in the AQ.

By limiting the total amount of bandwidth requested by premium traffic, the network administrators can guarantee that premium traffic will not starve the assured and best effort traffic.

To improve the required QoS we are supposed to solve the following tasks:

- Determining network congestion.
- Providing dedicated bandwidth for media flows.
- Providing individual network characteristics for given media flow.

At present there are two directions of investigating broadly classified under the monikers «scheduling algorithms» and «queue management algorithms» All of the router algorithms (scheduling and queue management) developed so far have been either able to provide fairness or simplicity to implement, but not both features simultaneously. We will take a step in the direction of bridging fairness and simplicity. Specifically, I have an idea to exhibit an active queue management algorithm, that could be simple to implement (since it doesn't requires state information) and could differentially penalize misbehaving.

The tasks to be proposed.

- Developing a simple, stateless algorithms that could achieve flow isolation and/or approximate fair bandwidth allocation and looking for a solution to the above problem in the context of the IP based networks. Also we need to find scheme that could differentially penalize «unresponsive» flows.

- Testing our research on high speed network based on the developed methods and algorithms. For evaluating the effect of the proposed methods as the base model for communication link we are going to use the mass servicing system with the heterogenous packets flows on *n* types, which get into the channel with the intensities $\lambda_1,...,\lambda_n$. Let V — the total link bandwidth, L_i — the average length of the type i packet. When the packets are of the same length and flows are simple, the average time delay for the type i packet will be calculated as following:

$$\tau_i = \frac{\sum_{j=1}^{n} \lambda_j L_j^2}{(V - \sum_{j=1}^{i-1} \lambda_i L_i)(V - \sum_{j=1}^{i} \lambda_i L_i)} + \frac{L_i}{V}, \quad i = 1, n .$$

All the mathematical calculations are supposed to be compared with the practical results of the algorithms.

Among the existing network technologies *Asynchronous Transfer Mode* (ATM) meets the QOS requirements. Nevertheless ATM technology is rather expensive and complicated, especially, for using in R&D networks.

REFERENCES

1. *Mieghem P. Van, De Neve H. and Kuipers F.A.* Hop-by-hop quality of service routing // Computer Networks. — October 2001. — **37**, № 3–4. — P. 423 – 497.
2. *Копачев А.Г.* Обзор методов и технологий по обеспечению качества предоставляемых услуг в компьютерных сетях передачи данных // Информатизация образования. — 2004. — № 4. — С. 59–70.
3. *Матрук А.А.* Качество обслуживания в компьютерных сетях // Информатизация образования. — 2005. — № 3. — С. 81–83.
4. *Stoica I., Zhang H.* LIRA: A model for service differentiation in the Internet // In proc. of NOSSDAV'98. — London: UK. — 1998.
5. *Floyd S., Jacobson V.* Link sharing and resource management models for packet networks // IEEE/ACM transactions on networking. — 1995. — **3,** № 4. — P. 365–386.
6. *Floyd S., Jacobson V.* Random early detection gateways for congestion avoidance // IEEE / ACM transaction on networking. — 1993. — **1**, № 4. — P. 397–413.

Received 15.03.2006