A Metric for Numerical Evaluation of the QoS of an Internet Connection

Falko Dressler

Department of Computer Science (Operating Systems), University of Erlangen-Nuremberg, Martensstr. 1, 91058 Erlangen, Germany

Today, the importance of providing quality of service (QoS) in the network for various types of real-time applications is widely accepted. Examples for such services are video communications and distributed co-working scenarios. Various mechanisms to provide or even to guarantee some minimum QoS values have been developed and others are current work in progress. Typically, the problem for an application and an end user is to estimate the currently available network quality. Even if QoS mechanisms are employed on all used parts of the network, it cannot be taken for granted that the prerequisites are met. Besides measurement mechanisms and analysis methods a metric is required allowing a numerical comparison of the available service quality and the demands of the particular application. This paper introduces a new calculation method which allows one to use measured QoS values and the application requirements to compute a numerical representation for the service quality. Additionally, the result of the computation enables a direct comparison of different communication paths for the same kind of service.

1. Introduction

The importance of providing or even guaranteeing a minimum quality of service (QoS) in the network is widely accepted. Especially real-time applications such as multimedia transmissions or remote control applications require much higher quality transmissions [8], [9]. Various working groups at the IETF (Internet Engineering Task Force) are working on concepts and principles which allow one to increase the service quality.

In order to allow one to measure the currently available quality of service in the network, single metrics have been created which include the definition of some measurement concepts for a particular network connection as well as on an end-to-end basis [13]. Additionally, tools and concepts have been developed to build a framework for such measurements. An example is the MQM (Multicast Quality Monitor) described by Dressler [10], [11].

In addition to the measurement a new question appeared: how to compare the QoS of different communication paths or end-to-end connections? Especially, if a meaningful and maybe automated comparison process is required, this action becomes even more difficult. An example for such a comparison is the selection of a multimedia server out of several sources which all contain the same content [12].

Within this paper a new approach for a possible solution is presented: a metric allowing a numerical evaluation of the quality of service of an internet connection. The here presented formula defines a calculation method which also deals with the question of accumulating single metrics as well as with the comparability of different metrics.

The paper is organized as follows: section 2. introduces some interesting quality of service parameters and corresponding metrics. The proposed calculation method is defined in section 3. Some application scenarios (section 4.) and a conclusion (section 5.) summarize this work.

2. Quality of Service Parameters and Single Metrics

In order to analyze the quality of service of a particular network connection, the single QoS parameters have to be defined and explained. To prevent the objective from being lost, only those parameters are described, which are imperative in most real-time communications. The list starts with the reachability, i.e. an end-to-end communication path must be available. Real-time multimedia applications depend predominantly on the delay of a transmission and require a low variation of the delay. Nearly every application depends on a low packet loss ratio.

The first step towards a framework for defining internet performance metrics has been provided by Paxson [19]. He distinguishes between analytically and empirically specified metrics. Another approach by Awerbuch et al. [5] describes a cost-sensitive analysis of communication protocols. Additionally, several documents have been written by the IETF in order to define a generally accepted framework for IP performance measurements [20].

In the following sections, the single QoS parameters are introduced including a simple calculation method allowing a numerical analysis of the single metric.

2.1. Reachability and Reliability

Connectivity between two end systems means that it is possible to transmit data between these two sites with at least a best effort behavior. Typical reliability measurements are based on this simple metric. A measurement methodology is defined in [17].

Using the results of reachability measurements over a period of time, the reliability of the network can be calculated. Therefore, reachability means connectivity at a certain point of time and reliability stands for the percentile reachability over a period of time. The resulting end-to-end quality can be expressed using the following formula:

$$q = \begin{cases} 1 & \text{connectivity is provided} \\ 0 & \text{else} \end{cases} \tag{1}$$

2.2. Delay

The delay, or more precisely the absolute delay, describes the latency between the transmission of a packet and its successful reception at the receiving site. In the case of a video transmission, a large but constant delay leads to a delayed playback, but does not reduce the overall quality of the transmission. For bidirectional conferences, for example, it has been shown that a delay larger than 200 ms reduces the interactivity dramatically [6].

One of the most important values for real-time multimedia communication is the one-way delay (OWD), because every transmission of audio or video signals is unidirectional from one host to another. Even in bidirectional video conferences, the one-way delay is very important due to its effects on each individual packet stream. Due to synchronization problems between the clocks of each client, the measurement of the one-way delay is a non-trivial task. For exact measurements, it is required that both clocks are highly synchronized. This topic was discussed, for example, by Mills [18] and Awerbuch [4]. A recommendation for the measurements of the one-way delay can be found in [2].

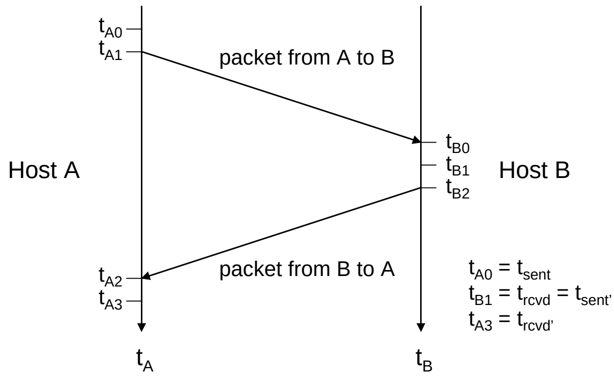

Figure 1. Delay between two Hosts

A schematic overview is given in Figure 1. Each host has its own independent time line (t_A, t_B). At time $t_{A0} = t_{sent}$ the application sends a packet to host B. A short delay is introduced by the application, the operating system, and the networking hardware, so that the packet is actually sent at t_{A1}. Host B receives the packet at t_{B0} and a timestamp is taken by the application at $t_{B1} = t_{rcvd}$ (there is an additional system depended delay between t_{B0} and t_{B1}). Using the timestamp t_{A0} it is possible to compute the one-way delay (Δt_{OWD}) as follows:

$$\Delta t_{OWD} = t_{rcvd} - t_{sent} \tag{2}$$

For applications which require a fast query-response behavior, e.g. video conferences and remote controls, the round-trip time (RTT) is an important value as well.

The discussion of Figure 1. can be continued to show the concepts of the RTT measurement. In thie case t_{B1} is also used as $t_{sent'}$ and is included in the response packet. At t_{A2} the packet is received by host A and the last timestamp is taken at $t_{A3} = t_{rcvd'}$. Using all these timestamps, it is possible to compute the round-trip time (Δt_{RTT}) as follows:

$$\Delta t_{RTT} = \Delta t_{OWD'} + \Delta t_{OWD} = t_{rcvd'} - t_{sent} \tag{3}$$

Because the measurement of the round-trip time depends only on $t_{rcvd'}$ and t_{sent}, only the clock of one host is involved. Therefore, no synchronization between both clocks is required. Because of this the measurement of the RTT is one of the standard measurements in computer networks. Additionally, the behavior of the network in terms of the available quality of service can be rated using this single parameter. The QoS of an end-to-end network connection regarding the delay can be expressed as follows:

$$q = \begin{cases} 1 & (d < d_{fav}) \\ d' & (d_{fav} \le d < d_{max}) \\ 0 & \text{else} \end{cases} \tag{4}$$

Whereas d is the measured delay, d_{fav} the optimum value for the particular application and d_{max} the worst case which can be at least tolerated. Any higher delay is intolerable by the application and, therefore, the end-to-end connection is useless. d' is a scaled down expression of d which is described later. This formula can be used for the OWD as well as for the RTT.

2.3. Jitter (Variation of the Delay)

The variation of the inter-arrival time of packets at the receiving site is known as the delay variation, also referred to as the jitter. There are two ways to measure the jitter.

The variation of the delay. Single delay measurements are taken over a period of time. The jitter is computed as the maximum variation around the mean delay value. Additionally, the a percentile, e.g. the 95th percentile, can be used instead of the mean value to remove spikes.

Typically, the variation of the delay is computed using the OWD measurements, because the jitter in the round-trip time is ostensibly meaningless for multimedia transmissions. Unfortunately, the same problem appears as in the calculation of the one-way delay. The clocks of all involved systems have to be highly synchronized.

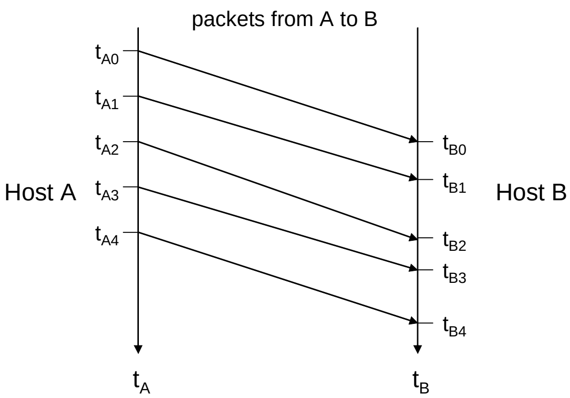

Figure 2. Jitter (Delay Variation)

An example is shown in Figure 2. Host A periodically sends packets to host B. It includes a timestamp in each packet for the OWD measurement at host B. The jitter can be calculated as:

$$\Delta t_{\text{jitter}} = \max_k \left(\left| \frac{\sum_{i=0}^{n} \Delta t_{\text{OWDi}}}{n+1} - \Delta t_{\text{OWDk}} \right| \right) \tag{5}$$

The variation of the interarrival time. Another method is to analyze the variation of the interarrival time of packets. Using a constant flow of packets with a well-defined inter-packet distance, the receiver measures the distances between the packets when it receives them. The jitter can be calculated as the maximum variation of the interarrival time over a period of time.

$$\Delta t_{\text{interarrival n}} = t_{\text{Bn}} - t_{\text{Bn-1}} \tag{6}$$

$$\Delta t_{\text{jitter}} = \max_k \left(\left| \frac{\sum_{i=1}^{n} \Delta t_{\text{interarrival i}}}{n} - \Delta t_{\text{interarrival k}} \right| \right) \tag{7}$$

Using this kind of jitter measurement, the problem of unsynchronized clocks disappears because only timestamps at the receiver are used for the computation. A metric for the IP packet delay variation was first published by Demichelis [7].

The calculation of a numerical value for the evaluation of the current jitter can be expressed using the same formula as described for the absolute delay:

$$q = \begin{cases} 1 & (j < j_{\text{fav}}) \\ j' & (j_{\text{fav}} \le j < j_{\text{max}}) \\ 0 & \text{else} \end{cases} \tag{8}$$

2.4. Packet Loss Ratio

The packet loss ratio is defined by the amount of packets which were lost in the last time slice. A measurement method is described in [3]. Packet loss is very common in the internet. Even if different encoding algorithms for multimedia content are available which allow the applications to deal with a small packet loss ratio, e.g. by using forward error correction mechanisms, the knowledge about the packet loss ratio is useful for several reasons [3]:

- most applications do not perform well if the packet loss reaches some threshold value
- excessive packet loss makes it difficult to support real-time applications
- the larger the number of packet loss, the more difficult it is for transport-layer protocols to sustain high bandwidths, especially when very large delay-bandwidth products must be supported.

Particularly in supporting multimedia streaming, it is not possible to use a reliable transport protocol for retransmissions of lost packets. Therefore, the packet loss ratio is a very important measure of the current behavior of the network. The quality of service of an end-to-end connection in terms of packet loss might be calculated using the following formula:

$$q = \begin{cases} 1 & (l < l_{\text{fav}}) \\ l' & (l_{\text{fav}} \le l < l_{\text{max}}) \\ 0 & \text{else} \end{cases} \tag{9}$$

In order to measure the packet loss ratio, a packet stream which includes sequence numbers is required. Typically, the real-time transport protocol (RTP, [21]) is used for such packet loss measurements which is also employed by most multimedia applications.

3. A Proposed Calculation Method

The goal of this section is to introduce a calculation method which allows one to compute numerical values for an end-to-end quality regarding the different application requirements. Three steps are required to calculate the desired result:

- calculation of the single end-to-end metrics
- weighting the metric values
- specification of a calculation method for the end-to-end quality.

All three steps are explained in more detail in the following. To achieve a more clear impression of the proposed technique and its application, some application scenarios are provided in the next section.

3.1. Calculation of the single end-to-end metrics

For each QoS parameter, the end-to-end quality has to be measured. This can be done using hop-by-hop measurement mechanisms but the most accurate results can be determined using end-to-end measurements. The choise of a particular measurement tool is not discussed here.

3.2. Weighting of single metric values

Calculation methods for quality of service specific metrics were discussed by Paxson [19]. A first thought on combining different parameters was presented by Kleinrock [16].

Wighting methods are necessary to evaluate the single measured metrics. In general, the measured value v has to be first scaled down using some scaling factors k, l so that $v' = k \times (v + l)$ is in the interval [0,1]. The factors k, l depend on the type of the metric as well as on the requirements of a particular service. Furthermore, given the demands of a particular service which may define a maximum for the metric v_{max} as well as a favored (optimum) value v_{fav}, the available quality m for a particular service can be defined as:

$$m = \begin{cases} 1 & (v < v_{fav}) \\ v' & (v_{fav} \le v < v_{max}) \\ 0 & \text{else} \end{cases} \tag{10}$$

Thus, the quality of a connection is defined as zero if no reasonable transmission is possible and as one if the demanded quality is available. If some metric value between the favored and the maximum value is acquired, the quality m is specified as the downscaled version of v, v'. In this case v' can be expressed using the maximum v_{max} and the optimum v_{fav} as follows:

$$v' = \frac{1}{v_{max} - v_{fav}}(v - v_{fav}) \tag{11}$$

The definition of the calculation method strongly depends on the metric type. For example, the calculation of the connectivity m can be achieved using the formula (which is obviously only a simplification of the general form shown in equation 10):

$$m = \begin{cases} 1 & \text{connectivity is provided} \\ 0 & \text{else} \end{cases} \tag{12}$$

Right now, we have a calculation method for single metrics generating a comparable numerical expression of the quality of a network connection. A high value of m, which might be 1, stands for a perfect service quality. A value between 0 and 1 describes an average quality and a value of 0 means that the service is not available.

3.3. Calculation of the End-to-End Quality

Typically, the end-to-end quality of a connection depends on more than a single metric. Thus, the quality of a connection c can be written as a vector of single weighted metrics m_i:

$$c = \begin{bmatrix} m_0 \\ m_1 \\ \dots \\ m_n \end{bmatrix} \tag{13}$$

A numerical representation of the overall connection quality c' is the product of the single values of c:

$$c' = m_0 \times m_1 \times \ldots \times m_n \qquad (14)$$

The higher the value of c' which is in the interval [0,1], the better the quality of the end-to-end connection is. If c' equals to zero, the connection cannot be used for the requested service.

Because the single values have been generalized to the interval [0,1], the resulting value of c' can become very low. Nevertheless, it allows a meaningful comparison of different end-to-end connections using the same metrics. This is shown in more detail in the following.

4. Application Scenarios

Two application scenarios are presented in this section: the verification of service level agreements (SLAs) and the proper selection of a suitable source for multimedia distributions.

4.1. Verification of SLAs

The basic reason for specifying metrics is to evaluate costs. Typically, service level agreements are used to specify a minimum service quality between ISPs and their customers and between different ISPs, respectively, which both parties agreed on. Typically, SLAs are set up at the borders of the different networks. Single metrics can be used to specify a minimum assured service quality. Today, no ISP is guarantees any end-to-end quality because. Nevertheless, first providers started to employ QoS mechanisms in their networks offering new service levels. Typically, such (expensive) services are called "premium" or "assured".

In general, there is still a discussion on the optimum definition of SLAs. In practice, a SLA is employed to define the behavior of a particular network connection. This type of SLA is much easier to specify and to control. In most research areas, the definition of the behavior of a network connection is demanded on an end-to-end basis [1], [19]. This also is the kind of SLAs demanded by the end user. Therefore, a new type of SLAs is required describing the assured end-to-end service quality.

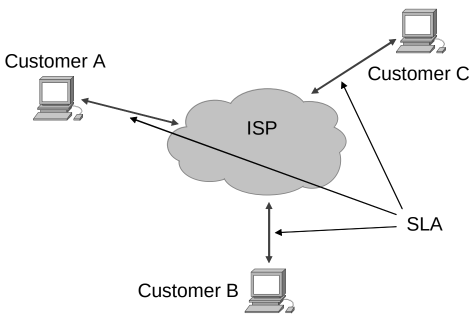

Figure 3. SLAs for a Video Conference

Basically, all the SLAs and the used metrics should be based on the demands of actually implemented services. An example is provided which introduces the mechanisms of performance and quality metrics as well as the concepts of service level agreements. As shown in Figure 3., three customers are connected to an ISP. They all completed the same SLA with the ISP. The contract is shown in the first two columns of table 1.

Table 1.

parameter	maximum value	optimum value	connection A-B	connection A-C
throughput	1Mbps	900kbps	27Mbps	1.2Mbps
delay	200ms	100ms	20ms	150ms
jitter	150ms	80ms	10ms	80ms
packet loss ratio	0.01	0.001	0.05	0.001

The values defined by the SLA must be compared to the measured values in order to have an instrument for verification of the promised service quality. Table 1 also shows the results of two measurements. The first one, which is labeled "connection A-B" between customer A and B, and a second one, which is labeled "connection A-C" between customer A and C.

Using the proposed formula, the numerical representation of the overall service quality for both connections can be calculated as follows:

1. Connection A - B

$m_{throughput}$ = 1 (measured value is better than the optimum)

m_{delay} = 1 (measured value is better than the optimum)

m_{jitter} = 1 (measured value is better than the optimum)

m_{loss} = 0 (measured value is worse than the minimum)

and therefore

$c_{AB} = m_{throughput} * m_{delay} * m_{jitter} * m_{loss} = 1 * 1 * 1 * 0 = 0$

which means that the connection from A to B cannot be used for the requested service.

2. Connection A - C

$m_{throughput}$ = 1 (measured value is better than the optimum)

m_{delay} = (150ms - 100ms) / (200ms - 100ms) = 0.5 (the measured value is in the interval between the maximum and the optimum and therefore, it is scaled down to the interval [0,1])

m_{jitter} = 1 (measured value is better than the optimum)

m_{loss} = 1 (measured value is better than the optimum)

and therefore

$c_{AC} = m_{throughput} * m_{delay} * m_{jitter} * m_{loss} = 1 * 1 * 0.5 * 1 = 0.5$

which means, that the connection from A to C allows one to use the requested service with an overall achieved quality of 0.5.

4.2. Selection of a suitable Source for Multimedia Distributions

Distribution and replication of content is a common technique to increase the availability and to allow some load balancing. Especially for mission-critical applications such as tele-medicine and tele-learning environments, a high availability is required. The problem of selecting a best suitable source for the forthcoming transmission shown in Figure 4. [14].

A client can choose one out of many available servers based on several criteria. First concepts used the availability of a particular server or even just the server load as a selection criterion. Newer approaches such as described by Dressler [12] include the currently available transmission quality from the server towards the client.

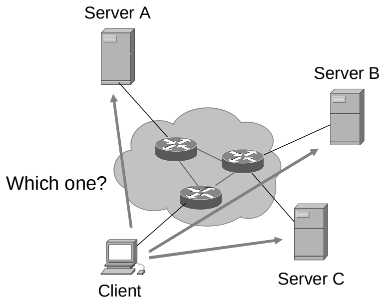

Figure 4. Problem Description

A short measurement of the current quality of service of the end-to-end connection from each available server towards the client is accomplished. The results can be numerically evaluated using the proposed calculation method and a best suitable server can be chosen automatically.

5. Conclusions

Summarizing the paper it can be said that it was possible to define a metric describing the overall quality of an end-to-end network connection. The resulting estimations from the calculation are directly comparable. Therefore, they allow an automated test and a complete analysis of network connections regarding their quality of service capabilities. The numerical representations of the achieved transmission quality are a direct result of single QoS measurements and the applications requirements.

The usability of the calculation method has been proven by single application scenarios. Differently than the former procedures, the proposed approach is able to cover a much broader spectrum of parameters of an end-to-end connection. Additionally, it permits the combination of several single metrics and it allows a direct comparison of the quality of multiple network paths.

To verify the correctness and the usability of the proposed metric, it is already employed for the selection process of sources of multimedia traffic out of distributed servers with replicated content. Based on the described calculation method, it is possible to get to better results while performing this selection process.

References

[1] A. Adams, R. Bu, R. Caceres, N. Duffield, T. Friedman, J. Horowitz, F. Lo Presti, S. Moon, V. Paxson, D. Towsley, "The use of end-to-end multicast measurements for characterizing internal network behavior," IEEE Communications Magazine, May 2000.

[2] G. Almes, S. Kalidindi, M. Zekauskas, "A One-way Delay Metric for IPPM," RFC 2679, IETF, September 1999.

[3] G. Almes, S. Kalidindi, M. Zekauskas, "A One-way Packet Loss Metric for IPPM," RFC 2680, IETF, September 1999.

[4] B. Awerbuch, "Complexity of network synchronization," Journal of the ACM (JACM), Volume 32 Issue 4, October 1985.

[5] B. Awerbuch, A. Baratz, D. Peleg, "Cost-sensitive analysis of communication protocols," Proceedings of the ninth annual ACM symposium on Principles of distributed computing, Quebec City, Quebec, Canada, August 1990.

[6] I. Busse, B. Deffner, H. Schulzrinne, "Dynamic QoS Control of Multimedia Applications based on RTP," Proceedings to First International Workshop on High Speed Networks and Open Distributed Platforms, St. Petersburg, Russia, June 1995.

[7] C. Demichelis, "Improvement of the Instantaneous Packet Delay Variation (IPDV) Concept and Applications," Proceedings to World Telecommunications Congress 2000, Birmingham, May 7-12, 2000.

[8] F. Dressler, "QoS considerations on IP multicast services," Proceedings of International Conference on Advances in Infrastructure for Electronic Business, Education, Science, and Medicine on the Internet (SSGRR 2002w), L'Aquila, Italy, January 2002.

[9] F. Dressler, "Advantages of VoIP in the German research network," Proceedings of 5th IEEE International Conference on High Speed Networks and Multimedia Communications (IEEE HSNMC 2002), Jeju Islands, Korea, July 2002, pp. 56-60.

[10] F. Dressler, "An Approach for QoS Measurements in IP Multicast Networks, MQM - Multicast Quality Monitor," Proceedings of Third International Network Conference (INC 2002), Plymouth, UK, July 2002.

[11] F. Dressler, "MQM - Multicast Quality Monitor," Proceedings of 10th International Conference on Telecommunication Systems, Modeling and Analysis (ICTSM10), vol. 2, Monterey, CA, USA, October 2002, pp. 671-678.

[12] F. Dressler, "An Approach to Select a Best Suitable Video Server;" Proceedings of International Conference on Advances in Infrastructure for Electronic Business, Education, Science, Medicine, and Mobile Technologies on the Internet (SSGRR 2003w), L'Aquila, Italy, January 2003.

[13] D. Ferrari, "Client Requirements for Real-Time Communication Services," RFC 1193, IETF, November 1990.

[14] M. Guo, M. Ammar, and E. Zegura, "Selecting among replicated batching video-on-demand servers," Proceedings of the 12th International Workshop on Network and Operating System Support for Digital Audio and Video (NOSSDAV 2002), Miami, Floria, USA, 2002, pp 155-163.

[15] M. Kanbara, H. Tanioka, K. Kinoshita, K. Murakami, "A Multicast Routing Algorithm for Multiple QoS Requirements," Proceedings of Third International Network Conference, INC 2002, Plymouth, UK, July 16-18, 2002.

[16] L. Kleinrock, "On Flow Control in Computer Networks," Proceedings of the International Conference on Communications (ICC), Vol. II, Toronto, Ontario, June 1978, pp. 2.7.21-2.7.25.

[17] J. Mahdavi, V. Paxson, "IPPM Metrics for Measuring Connectivity," RFC 2678, IETF, September 1999.

[18] D. Mills, "Network Time Protocol (Version 3) Specification, Implementation and Analysis," RFC 1305, IETF, March 1992.

[19] V. Paxson, "Towards a Framework for Defining Internet Performance Metrics," Proceedings of INET‘96, 1996.

[20] V. Paxson, G. Almes, J. Mahdavi, M. Mathis, "Framework for IP Performance Metrics," RFC 2330, IETF, May 1998.

[21] H. Schulzrinne, S. Casner, R. Frederick, V. Jacobson, "RTP: A Transport Protocol for Real-Time Applications," RFC 1889, IETF, January 1996.

[22] Z. Whang, J. Crowcroft, "Quality-of-Service Routing for Supporting Multimedia Applications," IEEE Journal on Selected Areas in Communications, Volume 14 Numer 7, 1996.