University of Würzburg
Institute of Computer Science
Research Report Series

The Network Utility Function: A Practicable Concept for Assessing Network Impact on Distributed Services

M. Fiedler, S. Chevul[1] and O. Radtke, K. Tutschku, and A. Binzenhöfer[2]

Report No. 355 April 2005

[1] Blekinge Institute of Technology, Karlskrona, Sweden
School of Engineering
Dept. of Telecommunication Systems
371 79 Karlskrona, Sweden
phone: (+46) 457-385000, fax: (+46) 457-27914
{markus.fiedler|stefan.chevul}@bth.se

[2] University of Würzburg, Germany
Institute of Computer Science
Department of Distributed Systems
Am Hubland, D-97074 Würzburg, Germany
phone: (+49) 931-8886641, fax: (+49) 931-8886632
{radtke|tutschku|binzenhoefer}@informatik.uni-wuerzburg.de

The Network Utility Function: A Practicable Concept for Assessing Network Impact on Distributed Services

M. Fiedler, S. Chevul
Blekinge Institute of Technology, Karlskrona, Sweden
School of Engineering
Dept. of Telecommunication Systems
371 79 Karlskrona, Sweden
phone: (+46) 457-385000, fax: (+46) 457-27914
{markus.fiedler|stefan.chevul}@bth.se

O. Radtke, K. Tutschku, and A. Binzenhöfer
University of Würzburg, Germany
Institute of Computer Science
Department of Distributed Systems
Am Hubland, D-97074 Würzburg, Germany
phone: (+49) 931-8886641, fax: (+49) 931-8886632
{radtke|tutschku|binzenhoefer}@informatik.uni-wuerzburg.de

Abstract

Based on the need for distributed end-to-end quality management for nextgeneration Internet services, this paper presents a ready-to-deploy quality assessment concept for the impact of the network on the service performance. The proposed Network Utility Function (NUF) combines the observed network utility at the inlet and the outlet. Thus, it captures the damping effect of the network onto user-perceived quality from an end-to-end perspective. As opposed to incomprehensible QoS parameters such as delay and loss, the NUF is highly intuitive due to its mapping to a simple value between 0 and 1. We demonstrate the capabilities of the proposed concept for a special NUF, the Throughput Utility Function (TUF) by realistic simulation.

Keywords: Keywords User-perceived QoS; End-to-End QoS; Performance Monitoring; Utility Function; Throughput

1 INTRODUCTION

The Internet has become the standard communication network for almost any network application. Its success results from its robustness, e.g. to choose alternative routes on link failures, as well as from the *end-to-end (E2E) concept*. The E2E concept allows the end-nodes to throttle the data flow between them and to adapt it to varying load conditions. However, the E2E concept also has disadvantages. The network, in general, does not know about the requirements and the actions of the end-nodes and thus can hardly adapt to these needs.

In addition, an increasing number of applications require strict network performance. *Quality of Service (QoS)* defines the "degree of conformance of the service delivered to a user by a provider in accordance with an agreement between them" [1]. As any other system, network applications in the Internet occasionally experience service quality problems. The observed problems may be caused by the implementation of the application or by interconnecting networks. While *network QoS* captures the impact of non-ideal E2E transportation, the *application QoS* perceived by the user comprises both, the performance of the application and of networks. Depending on the task a user is carrying out, problems with applications and the underlying networks are felt to be more or less annoying [2]. Users rate the service quality, application QoS as well as network QoS, in a subjective and individual way.

According to [3], user satisfaction typically depends on the *expected QoS*, the *user-perceived QoS* and the *applied pricing model* [4]. The expected QoS is influenced by the context of usage and the QoS parameters of interest such as response times [2] or video picture quality [5]. Whereas the user-perceived QoS is influenced by the system's performance and load as well as the perceived usability. The latter reflects the way the system presents itself to the user, e.g. through bad response times or picture quality. In general, users tolerate at least some service quality degradation, but as soon as the service quality falls below a certain *acceptance threshold*, many of them complain either explicitly, e.g. by shouting at the provider, or implicitly, e.g. by giving up using the service. The latter can have a quite crucial impact on the economical situation of the service provider [6].

Thus, for a service provider, it is important to find out about such acceptance thresholds and their correlation with problematic states of applications and/or networks [3]. Appropriate quality metrics are robust, useful from the viewpoint of the application and universal, i.e. usable for any kind of service. More important, they should be simple and only rely on a few, easily tunable parameters. *Utility functions* enable us to relate the state of application and network to end-user perceived QoS and user satisfaction. They are used for rate control and resource allocation [7, 8, 9]. In this paper, we are particularly interested in the damping impact of the network QoS onto the utility function. This impact is captured by the proposed *Network Utility Function* (NUF).

QoS monitoring has to be performed in the same E2E way the user rates the service quality in the Internet. If applied at all, QoS monitoring and provisioning is usually carried out by rather centralized entities and only in those parts of the network where a provider is responsible for. A coordination of those activities among different administrative domains is rarely achieved. A *central QoS monitoring* typically results in additional, complex entities at the provider. A *decentralized QoS monitoring*, however, located on the user's end system and thus "observing" the QoS from the same perspective as the user, may avoid these disadvantages. Furthermore, the use of a distributed, self-organizing QoS monitoring architecture as suggested in [10] will reduce capital and operational expenditures (OPEX and CAPEX) of the operator since fewer entities have to be installed and operated. The architecture can inform a network manager about degraded E2E performance in terms of NUF values. It will complement today's solutions for central fault and performance management such as HP OpenView [11] by providing well defined interfaces. From the user's point of view, such an approach implements a highly desirable one-stop service concept [3].

In general, QoS can be characterized in terms of speed, accuracy and reliability [12]. Perceived *throughput* is a speed-related parameter that is important both for streaming applications such as videoconferencing, requiring a certain speed, and elastic applications such as file transfers, for which it impacts the download time [13]. Reference [14] shows how to use passive, unsynchronized throughput measurements as a *bottleneck indicator*. Such measurements form the base for the *Throughput Utility Function* (TUF), which is an example of the NUF concept. Applied within the recently presented self-organizing infrastructure [10], we have a comparably simple and user-friendly, yet powerful concept at hand to reveal the quality of network connectivity on a scale from 0 to 100 (%).

Alternative QoS evaluation concepts are suggested in [13, 15]. [13] proposes a so-called "fun factor" for elastic traffic such as TCP that reacts upon impairments by reducing its sending speed. It compares file download times with their ideal values as if the installed access link speed could have been fully used, which is an important point of user concern. The approach in [15] presents a similar type of calculation of the E2E "overall connection quality" as suggested in Section 2. However, the framework defines metrics based on connectivity, delay, jitter and packet loss, but hardly comments on the applicability of the concept to real networks.

The remainder of this paper is organized as follows. Section 2 presents the utility-function-based NUF concept and Section 3 concretizes it in form of the TUF. Section 4 illustrates the TUF using trace-driven OPNET simulations in realistic network scenarios. Section 5 provides conclusions and outlook.

2 THE NETWORK UTILITY FUNCTION (NUF) CONCEPT

In this section, we describe an approach of capturing the impact of a network on user-perceived QoS. We build this approach on the notion of an *utility function* [3, 7, 8, 9]. In Section 2.3.5 of [3] is stated: "In order for the rational players of a game – or an auction – to get what they really want, they need a way to express their relative preferences for the various outcomes of the game. To this end, an appropriate mathematical tool is used; namely the utility function. This is a function that reflects the ordering of user preferences regarding the various outcomes of the game by assigning to each outcome a value."

Let U_{in} denote the value of the utility function at the sender, i.e. at the inlet of the network. The performance-damping impact of the network is captured by the *network utility function* U_{Netw} providing an E2E view. The utility function at the receiver, i.e. at the outlet of the network, becomes

$$U_{\text{out}} = U_{\text{Netw}} \cdot U_{\text{in}} \,. \tag{1}$$

The parameters U_{in}, U_{out} and U_{Netw} range from 0 in the worst case to 100 % in the best case. Compared to technical QoS parameters such as delay (variation) and loss, the network utility function is rather intuitive for users, providers and operators [3]. Users can rate perceived service quality on a scale between 0 and 100 and define thresholds for unacceptability [5]. Service providers and operators can use those values to take measures against the quality problems, e.g. search for bad network conditions; reconfigure service and network; compensate affected users; or shut down the service for maintenance. Percentages are also highly appreciated as quality indicators in business processes, e.g. for demonstrating successful quality assurance in service provisioning [16]. Naturally, the network utility function U_{Netw} reaches its best value of 100 % if no network was present or if the network behaved perfectly, which means that the sent data streams are received instantaneously with unchanged inter-packet times. In that

Table 1: Example of loss-defined service classes and possible utility functions.

Service class	Loss ratio	$\max\{$utility function$\}$	
		Alternative 1	Alternative 2
Gold	0.5 %	95 %	90 %
Silver	1.0 %	90 %	80 %
Bronze	2.0 %	80 %	60 %

case, the user quality perception is that of the application quality alone ($U_{\text{out}} = U_{\text{in}}$). However, a lower value of U_{Netw} indicates a disadvantageous change of traffic properties between the corresponding endpoints. In the worst case, the perceived utility U_{out} reaches zero, which can be related to a very badly behaving network ($U_{\text{Netw}} \to 0$), a very bad service quality already on the sender side ($U_{\text{in}} \to 0$) or a combination of both.

For a service of interest, U_{Netw} should capture the network problems of interest in a way that matches changes in user perception such that the same rating applies on both sender and perceiver side. Table 1 shows an example of loss-defined service classes from the ETSI TIPHON project and extends it by examples of possible values of utility functions. The utility functions can capture several parameters taking influence on the service quality. In case these influences are rather independent of each other, we can define utility functions $U_{\text{Netw},i} \in [0, 1]$ and apply the following relationship (cf. also [15]):

$$U_{\text{Netw}} = \prod_i U_{\text{Netw},i} \, . \tag{2}$$

In the following, we discuss a concrete application of this concept.

3 THE THROUGHPUT UTILITY FUNCTION (TUF)

Disturbances of the throughput are captured by the bottleneck indicator concept introduced in [14]. At each endpoint, n throughput values are measured during an observation window of ΔW. Each throughput value denotes the average bit rate perceived during a rather small averaging interval ΔT, typically between 100 ms and 1 s. From the comparison of throughput histograms at sender and receiver with a throughput resolution of ΔR, we can derive information about whether the network path in-between sender and receiver appears as a shaping or shared bottleneck. As opposed to active measurements, no probing traffic is injected into the network; the only extra traffic stems from the exchange of measurement results (a couple of bytes after each observation window).

A condensed form of the bottleneck indicator consists of the following parameters (and their dependencies):

- the average throughput at the sender (network inlet), $m_{\text{in}}(\Delta W)$;
- the standard deviation of the throughput at the sender, $s_{\text{in}}(\Delta W, \Delta T)$;
- the average throughput at the receiver (network outlet), $m_{\text{out}}(\Delta W)$;

- the standard deviation of the throughput at the receiver, $s_{\text{out}}(\Delta W, \Delta T)$.

To simplify the notation, the dependencies on ΔW and ΔT are ommitted from now on. Changes of these parameters between both ends reflect the following network problems (in the order of severeness):

1. $m_{\text{out}} < m_{\text{in}}$: *Outstanding traffic* at the end of the observation interval, which means that an amount of data $(m_{\text{out}} - m_{\text{in}})\Delta W$ may arrive in a later observation interval or might have been lost;

2. $s_{\text{out}} > s_{\text{in}}$: *Increased burstiness* of the traffic, which means that the distribution of the throughput is more spread at the receiver than at the sender e.g. due to interaction with other traffic [14];

3. $s_{\text{out}} < s_{\text{in}}$: *Decreased burstiness* of the traffic, which means that the distribution of the throughput is condensed because of traffic shaping [14].

As changes of average (m) and standard deviation (s) are orthogonal effects, we propose in analogy to (2)

$$U_{\text{Netw}} = U_m \cdot U_s \,. \tag{3}$$

The *m-utility function* $U_m = f(m_{\text{in}}, m_{\text{out}})$ reflects the outstanding traffic. In the following, we assume a simple linear dependency on the *loss ratio* $\ell = \max\{1 - \frac{m_{\text{out}}}{m_{\text{in}}}, 0\}$ as follows:

$$U_m = \max\{1 - k_m \ell, 0\} \tag{4}$$

where k_m denotes the degree of utility reduction. The m-utility function approaches zero as ℓ approaches $1/k_m$. The max-operator prevents U_m from becoming negative in case $\ell > 1/k_m$. In Table 1, alternative 1 corresponds to $k_m = 10$ and alternative 2 to $k_m = 20$, respectively. The study of elastic traffic in [13] (Figure 2) indicates $k_m \simeq 1$. Figure 1 displays some utility functions.

The *s-utility function* $U_s = g(s_{\text{in}}, s_{\text{out}})$ captures the change in *burstiness*, where an unchanged value $s_{\text{out}} = s_{\text{in}}$ implies maximal utility. In general, a reduction in burstiness when passing a network (e.g. because of a low-capacity access link) is less critical than a growth. Reduced burstiness means that traffic flows more regularly through the network, which in turn implies the need of less spare capacity in order to maintain the desired QoS level [17]. A smoothed data stream might also be of advantage for the receiving application. We consider a rating dependent on the sign of the change $s_{\text{out}} - s_{\text{in}}$ by defining two different parameters $k_s^{\pm}$. An approriate linear dependency on the difference $\sigma = \frac{s_{\text{out}} - s_{\text{in}}}{s_{\text{in}}}$ is given as follows:

$$U_s = \begin{cases} \max\{1 - k_s^{+}\sigma, 0\} & \text{for } s_{\text{in}} < s_{\text{out}} \\ 1 & \text{for } s_{\text{in}} = s_{\text{out}} \\ \max\{1 + k_s^{-}\sigma, 0\} & \text{for } s_{\text{in}} > s_{\text{out}} \end{cases} . \tag{5}$$

Here, the parameter k_s^{+} reflects the decrease of U_s when the standard deviation doubles, while k_s^{-} does the same for a vanishing standard deviation. For instance, $k_s^{+} > k_s^{-}$ implies a stronger utility decrease in case of rising burstiness as discussed above.

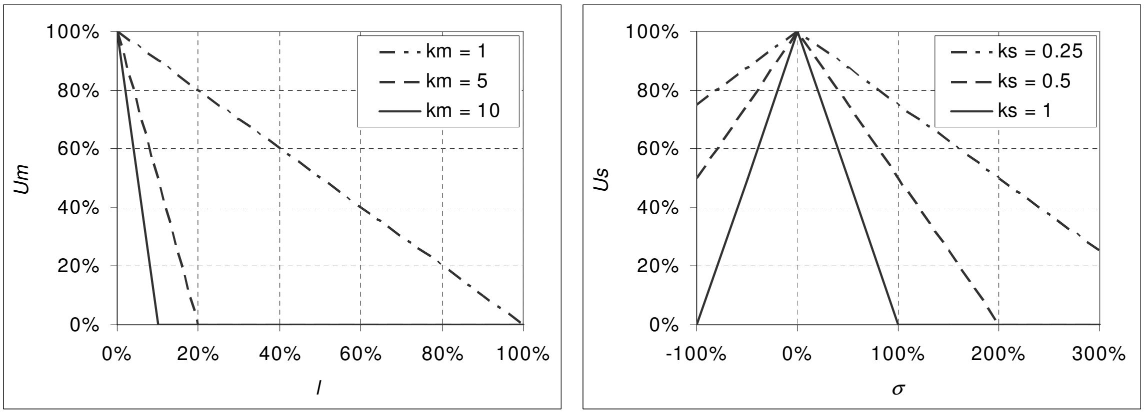

Figure 1: Illustrations of m-utility functions (left) and s-utility functions (right).

4 ILLUSTRATIVE EXAMPLE

We demonstrate the feasibility of the proposed concept by considering a videoconferencing service running over a heterogeneous network as shown in Figure 2, which was simulated using the well-known and established simulation software OPNET [18]. Two computers running the videoconference application (namely ViCo2 and LAN) are connected to a company LAN (100 Mbps Ethernet), a third one is connected to the Internet via Ethernet (ViCo1) and another one via ADSL (512 kbps upstreams/1536 kbps downstreams), respectively. The measurement points are to be found on local Ethernet links connecting the computers with their corresponding access routers as it was the case if wiretaps were used in a real experiment [14]. The company is connected to the Internet by a 2 Mbps digital subscriber line (DSL). This bottleneck link is additionally loaded by TCP traffic consisting of FTP, HTTP and SMTP and flowing from “Traffic Server” towards “Traffic Client” in parallel to the videoconference streams ViCo1→ViCo2 and ADSL→LAN. In the reverse direction, the two videoconference streams ViCo2→ViCo1 and LAN→ADSL share the bottleneck with each other and rather little TCP acknowledgement traffic. As the current OPNET videoconference source models produce constant bit rate traffic, real videoconference UDP packet traces stemming from MS Netmeeting conferences [19] were used as input for the simulator. However, the TCP traffic sources were modeled by OPNET itself in order to take care of the correct TCP feedback [20].

As we are aiming at demonstrating a concept rather than at carrying out a quantitative study, we confine ourselves to one observation window $\Delta W = 1$ min. We apply an averaging interval of $\Delta T = 100$ ms [14]. Given these settings, the traces displayed an average bit rate of roughly 450 kbps and a maximal bit rate of about 1200 kbps during 100 ms. The latter implies a risk of short-lived overload situations due to traffic variability. However, both videoconferences together load the bottleneck link rather modestly by $\sim$45 %. Downstreams, i.e. towards the LAN, the remaining capacity is consumed by the TCP-based applications.

In order to disturb the videoconferences and to visualize the changes in utility functions, the OPNET IP cloud was configured such as to introduce a selected packet loss ratio (LR = 1 or 5 %) and/or *additional* packet delays according to given distributions, such as uniform

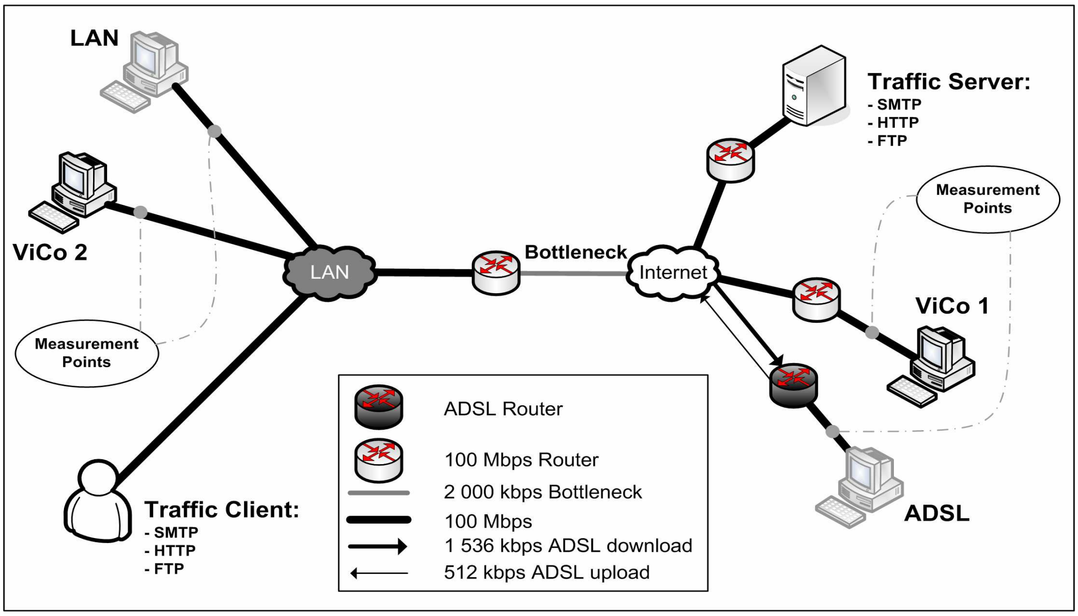

Figure 2: Simulated scenario.

Table 2: Values of utility functions in bottleneck downstream direction (incl. TCP data traffic).

Conference	AD	LR [%]	ℓ [%]	U_m [%]	σ [%]	U_s [%]	U_{Netw} [%]
ADSL→LAN	0	0	0.4	96	−25.5	87	83
ADSL→LAN	0	1	1.3	87	−25.3	87	76
ADSL→LAN	0	5	5.4	46	−24.8	88	40
ViCo1→ViCo2	0	0	0.1	99	−4.4	98	97
ViCo1→ViCo2	0	1	0.9	91	−5.5	97	88
ViCo1→ViCo2	0	5	4.7	53	−5.7	97	52

between 0 and 100 ms (AD = U) and exponential with a mean of 50 ms (AD = E). The undisturbed cases are denoted by LR = 0 % and AD = 0, respectively. As utility reduction parameters, we choose $k_m = 10$, $k_s^+ = 1$ and $k_s^- = 0.5$.

Let us first focus on the bottleneck downstream direction from Internet towards the LAN. Table 2 shows corresponding values of utility functions. The corresponding ratio ℓ almost matches the configured packet loss probability in the IP cloud. In the LR = 0 % cases, some traffic still seems to be on its way at the end of the observation period. For LR = 5 % implying $U_m \simeq 0.5$, the utility sinks to about half of its original value. Overall, the ADSL→LAN conference perceives worse utility functions than the ViCo1→ViCo2 conference. This is due to heavy shaping on the 512 kbps ADSL uplink loaded on average by 450 kbps and at maximum by 1200 Mbps during 100 ms. From Table 2, we observe a relative reduction of the standard deviation of about 25 % rather independently of the induced loss in the IP cloud, which leads to values of the s-utility function of about 87 %. The ViCo1→ViCo2 conference is shaped rather modestly due to the interaction with the ADSL→LAN conference and the

Table 3: Values of utility functions in bottleneck downstream direction in case of additional delays.

Conference	AD	LR [%]	ℓ [%]	U_m [%]	σ [%]	U_s [%]	U_{Netw} [%]
ADSL→LAN	U	0	0.3	97	-18.9	91	88
ADSL→LAN	U	5	5.1	49	-16.5	92	45
ADSL→LAN	E	0	0.5	95	58.9	41	39
ADSL→LAN	E	5	5.0	50	41.0	59	30
ViCo1→ViCo2	U	0	0.2	98	-1.1	100	98
ViCo1→ViCo2	U	5	5.2	48	-0.5	100	48
ViCo1→ViCo2	E	0	0.1	99	66.8	33	33
ViCo1→ViCo2	E	5	4.9	51	51.3	59	30

Table 4: Values of utility functions in bottleneck upstream direction (TCP ACKs only).

Conference	AD	LR [%]	ℓ [%]	U_m [%]	σ [%]	U_s [%]	U_{Netw} [%]
LAN→ADSL	0	0	0.0	100	-6.0	97	97
LAN→ADSL	0	1	1.0	90	-6.6	96	87
LAN→ADSL	0	5	4.8	52	-7.8	96	50
ViCo2→ViCo1	0	0	0.0	100	-8.0	98	96
ViCo2→ViCo1	0	1	1.0	90	-8.1	97	87
ViCo2→ViCo1	0	5	4.9	51	-10.5	97	49

TCP background traffic in the bottleneck.

We now add some additional delay to traffic crossing the IP cloud as described above. Table 3 presents the corresponding impacts in comparison to Table 2. For the ADSL→LAN conference, the effect of the shaping is slightly reduced by a uniform distribution of additional delay (U). However, an exponential delay (E) worsens the s-utility functions considerably to values between about 40 and 60 %. Obviously, the shaping has turned into the reverse: The throughput distribution is much broader at the outlet of the network than it was without the disturbance. Altogether, we yield throughput utility function values of 30 to 40 %. We obtain a similar behavior for the ViCo1→ViCo2 conference. While the uniform distribution of additional delay (U) implies almost neutral behavior with regards to the s-utility function, the exponential distribution (E) again worsens the values of the utility function considerably – exactly as in the ViCo1→ViCo2 case.

We now consider the bottleneck upstream direction in which the two videoconferences only interfere with the TCP acknowledgements. Table 4 shows the corresponding results. Both conferences perceive similar performance (slight shaping), as the ADSL downlink speed of 1536 kbps is no bottleneck for a single conference. The values of the throughput utility function are dominated by the loss. The shaping perceived by the ViCo2→ViCo1 conference is slightly larger than that in the reverse direction, cf. Table 2. Obviously, the TCP traffic affects the values of the utility functions of the UDP videoconferencing streams merely to a little extent. This maps with the common observation that UDP traffic displaces TCP traffic [21].

5 CONCLUSIONS AND OUTLOOK

We have described and demonstrated a practicable concept for distributed end-to-end QoS monitoring and assessment on service level, the Network Utility Function (NUF). The NUF relates utility functions at network inlet and outlet and thus captures the damping effect of the network onto user-perceived quality. We investigated a special NUF related to throughput changes, the Throughput Utility Function (TUF). The TUF captures changes of throughput averages (outstanding or lost traffic at the end of an observation period) and standard deviations on rather short time scales (delay variations), which was demonstrated by trace-driven OPNET simulations in a realistic network scenario.

Until now, we assumed a certain (linear) TUF dependency on changes of averages and standard deviations. A next step would be to determine threshold values regarding user acceptance of the service. In case the value of the throughput utility function drops below such a threshold, a QoS alarm should be issued. This can happen by SNMP trap messages towards a Network Management System (e.g. [11]), followed by control actions aiming at improving the QoS experience. Alternatively, the NUF or TUF can be determined through comparative experiments between the service quality at the inlet and at the outlet of the network, given a certain network impairment (e.g. loss or throughput variation) and followed by determining thresholds for issuing QoS alarms as described above. In practice, these steps should be carried out in cooperation with service and network providers.

ACKNOWLEDGEMENTS

The authors would like to thank the Network of Excellence *EuroNGI* for sponsoring this work through the *AutoMon* project and the colleagues from the EuroNGI work package WP.JRA.6.1 *Quality of Service from the user's perspective and feed-back mechanisms for quality control* for valuable input and discussions.

References

[1] ITU-T, "Recommendation E.860: Framework of a service level agreement," June 2002.

[2] A. Bouch, A. Kuchinsky, and N. Bhatti, "Quality is in the eye of the beholder: Meeting user's requirements for Internet Quality of Service," Tech. Rep. HPL-2000-4, HP Laboratories Palo Alto, January 2000.

[3] M. F. (ed.), "EuroNGI Deliverable D.WP.JRA.6.1.1 – State-of-the-art with regards to user-perceived Quality of Service and quality feedback," May 2004. URL: http://www.eurongi.org/, http://www.ats.tek.bth.se/eurongi/dwpjra611.pdf.

[4] A. Bouch and M. Sasse, "It ain't what you charge it's the way you do it: a user perspective of network QoS and pricing," in *Proceedings of IM'99, Boston MA*, 1999.

[5] H. Hlavacs, J. Lauterjung, G. Aschenbrenner, E. Hotop, and C. Trauner, “QoS measurement,” Deliverable D-3400, CODIS, November 2003. URL: http://www.ani.univie.ac.at/∼hlavacs/.

[6] Z. Research, Inc., “The economic impacts of unacceptable web-site download speeds,” white paper, Zona Research, Inc., April 1999.

[7] F. P. . Kelly, “Charging and rate control for elastic traffic (corrected version),” *European Transactions on Telecommunications*, vol. 8, pp. 33–37, Jan. 1997.

[8] R. J. La and V. Anantharam, “Utility-based rate control in the Internet for elastic traffic,” *IEEE Transactions on Networking*, vol. 10, pp. 272–285, April 2002.

[9] M. Dramitinos, G. D. Stamoulis, and C. Courcoubetis, “Auction-based resource allocation in umts high speed downlink packet access (HSDPA),” in *Proceedings of the First EuroNGI Conference on Traffic Engineering (NGI 2005)*, (Rome, Italy), April 2005.

[10] A. Binzenhöfer, K. Tutschku, B. auf dem Graben, M. Fiedler, and P. Carlsson, “A P2P-based framework for distributed network management,” Tech. Rep. 351, Universität Würzburg, January 2005.

[11] HP OpenView Management Software. URL: http://www.openview.hp.com/.

[12] ITU-T, “Recommendation I.350: General aspects of quality of service and network performance in digital networks, including ISDNs.” URL: http://www.itu.ch/.

[13] J. Charzinski, “Fun Factor Dimensioning for Elastic Traffic,” in *Proceedings of the ITC Specialist Seminar on Internet Traffic Measurement, Modeling and Management, Sept. 18–20, Monterey, USA*, 2000.

[14] M. Fiedler, K. Tutschku, P. Carlsson, and A. Nilsson, “Identification of performance degradation in IP networks using throughput statistics,” in *Providing Quality of Service in Heterogeneous Environments. Proceedings of the 18th International Teletraffic Congress (ITC-18)* (J. Charzinski, R. Lehnert, and P. Tran Gia, eds.), (Berlin, Germany), pp. 399–407, September 2003.

[15] F. Dressler, “A metric for numerical evaluation of the QoS of an Internet connection,” in *Providing Quality of Service in Heterogeneous Environments. Proceedings of the 18th International Teletraffic Congress (ITC-18)* (J. Charzinski, R. Lehnert, and P. Tran Gia, eds.), (Berlin, Germany), pp. 1221–1230, September 2003.

[16] T. Ingvaldsen, T. Jensen, H. Kjønsberg, H. Abrahamsen, and E. Smørdal, “On key performance indicators for end-to-end performance in cellular networks,” in *Proceedings of the Seventeenth Nordic Teletraffic Seminar (NTS 17)*, (Fornebu, Norway), pp. 37–48, August 2004.

[17] M. Fiedler and R. G. Addie, "Verification and application of a second-order scale symmetry for queueing systems," in *Teletraffic Engineering in a Competitive World. Proceedings of the 16th International Teletraffic Congress (ITC-16)* (P. Key and D. Smith, eds.), (Edinburgh, UK), pp. 807–816, June 1999.

[18] OPNET. URL: http://www.opnet.com/.

[19] N. Microsoft. URL: http://www.microsoft.com/windows/netmeeting/.

[20] Å. Arvidsson and P. Karlsson, "On traffic models for TCP/IP," in *Teletraffic Engineering in a Competitive World. Proceedings of the 16th International Teletraffic Congress (ITC-16)* (P. Key and D. Smith, eds.), (Edinburgh, UK), pp. 447–456, June 1999.

[21] A. Veres, Z. Kenesi, S. Molnár, and G. Vattay, "Tcp's role in the propagation of self-similarity in the internet," *Computer Communications, Special Issue on Performance Evaluation of IP Networks*, vol. 26, pp. 899–913, May 2003.