Optimal Bandwidth Allocation in IP network; the case of QoS-sensitive user utility functions

Piotr Paulski[*], Mariusz Kamola[*†]

[*] **Inst. of Control and Comp. Eng., Warsaw Univ. of Technology, ul. Nowowiejska 15/19, 00-665 Warsaw, Poland**
P.Paulski@stud.elka.pw.edu.pl, mkamola@ia.pw.edu.pl
[†] **NASK (Research and Academic Computer Network), ul. Wąwozowa 18, 02-796 Warsaw, Poland**
Mariusz.Kamola@nask.pl

Keywords: quality of service, congestion and admission control, service and QoS pricing

Abstract

The problem of social utility maximization for transmission services over a shared network link is addressed. The standard approach to such case, based on problem decomposition and optimal bandwidth allocation through adequate price setting, is extended towards a case where packet delay impacts user utility. A simple yet realistic class of functions modeling utility is proposed. A price-driven allocation algorithm is proposed, able to allocate resources optimally and in socially acceptable way. Simulation tests show that the algorithm operates correctly and efficiently.

1. INTRODUCTION

The increasing demand for bandwidth in the Internet poses a problem of efficient allocation of resources to network users on links where bottlenecks occur. Usually, this happens rather on Internet edges, where bandwidth is a scarce, highly priced, highly utilized and non-expandable resource – like in case of wireless access or peering links. Simultaneously, managing QoS differentiation there is costly in terms of human effort, processing power of network equipment and prices of resource allocation management software. Therefore, those links remain mostly best-effort ones. On the other hand, nowadays QoS does present a significant factor for a user when it comes to service pricing.

In this paper a concept is presented for bandwidth allocation to users that takes into account the service utility based on both the allocated bandwidth and the experienced transmission delay in a single, shared link. We assume that no mechanisms for QoS differentiation are in force except for sheer connection admission control –so all users perceive the same transmission delay. Users' preferences are expressed in the form of individual utility functions of two arguments: the individually allocated bandwidth and the commonly experienced transmission delay.

Below are given guidelines for the construction of the allocation algorithm being presented:

- Pricing scheme should be simple and uniform for all users, with preference for equal price for each bandwidth unit allocated;
- User utility function is considered private; this should be respected as much as possible by the allocation algorithm;
- The algorithm must not impose too many requiements on the way traffic mixes and influences delay – it should rather adapt to the current link state.

This paper has the following structure. Section 2 presents the research context. In Section 3 the problem of optimal resource allocation is stated formally, followed by the discussion on necessary assumptions and propositions. Section 4 presents the way the algorithm is implemented, with emphasis on the user's probable attitude towards it. Section 5 gives simulation results for the algorithm implementation. Section 6 concludes the work.

2. THE BACKGROUND AND RELATED WORK

One of fundamental resource allocation schemes was presented by Low and Lapsley in [11], where social utility was maximized by a bandwidth pricing algorithm. The optimization problem for just one link case was as follows:

$$\max_{m_i \le x_i \le M_i,\ i=1,\dots,n} \sum_{i=1}^{n} u_i(x_i), \qquad \text{subject to} \quad \sum_{i=1}^{n} x_i \le c \qquad (1)$$

Here, the maximum total utility was sought, being the sum of utilities $u_i(x_i)$ that each user i associates with being given the bandwidth x_i. Each user's utility is strictly concave and increasing for $x \in \langle m_i, M_i \rangle$, the interval of interesting allocations for the user i. The problem is also subject to resource allocation constraint, the link capacity c.

The dual of such problem can be decomposed into the optimization task of the coordinator that offers a Lagrange multiplier, a uniform price p per bandwidth unit, to local network users. Given such price, each user decides upon the amount of bandwidth bought so that his objective,

$u_i(x_i)$-px_i, is maximized. The role of the coordinator is to execute control so that the resource constraint shall not be violated.

The above scheme does really maximize the social utility, making the users feel they can make the decisions on their own, while in practice those decisions are determined, although indirectly, by appropriate price setting by the coordinator. An equivalent algorithm to the one by Low and Lapsley, was proposed in [9]: the users report periodically their budget for transmission in the coming period, and they are allocated bigger or lesser part of the link, depending on how high their offer was ranked by a sort of a central allocation algorithm.

The idea of utility-based resource allocation problem formulated in (1) has inspired many research activities. For example, in [7] the problems of resource allocation and optimal routing for flows are addressed jointly, which leads to a mixed integer programming problems. Appropriate heuristics are proposed, and reported to work properly. Another research direction focuses on the issue of resource constraints, trying to make them more adequate to what actually is going on in the real network: traffic variability and statistical multiplexing. It turns out [10] that resource allocation can be based as well on effective bandwidth theory for data flows. Therefore, QoS-related issues count more and more.

An interesting variation of the discussed seminal algorithm [11] is given in [4]. First, QoS is handled by utility values reduced by a factor proportional to the product of the observed delay τ, and the allocated bandwidth x: $u^*(x,\tau) = u(x) - \alpha\tau x$. The price communicated to users that guarantees attaining the optimum is the marginal increase of the delay with the increase of the network load (whoever of users causes it). The proposed algorithm assumes all customers to be price takers.

Transmission delay may be treated even more generally than a utility component: it may be utilized as a coordination signal itself [1]. In such case the networking infrastructure nature plays the role of a pricing algorithm as the extra queuing delay can serve as a good indicator of link capacity being used up. The users react automatically by diminishing their appetite for bandwidth. Such feature has been incorporated into a TCP congestion control scheme, and proved to be promising in a simulation environment.

It must be emphasized that none of the above approaches is robust to user collusion, or other forms of cheating, like artificially reducing demand in order to take advantage of diminishing bandwidth pricing. Usually, an assumption is made that the number of users is sufficiently big so that an individual decision influence on link state is negligible. This can be considered a rather slippery argument, especially that in a region of high link utilizations one's decision may count a lot. A proper addressing of users' decision interactions should be a game-theoretic then. There goes an extensive research work on auction design (cf. e.g. [8]) with the aim to construct such market rules that the optimal strategies for individual users are leading to a solution desirable by the game organizer. Reformulation of the game-theoretic approach with proper regard for possible collusions remains still, however, a relatively unexplored research field.

3. PROBLEM DEFINITION AND DISCUSSION

The algorithm presented in this paper is firmly based on the approach presented in [11], with several additions. In order to make the decomposition possible, and to guarantee convergence, several assumptions must be made about the way the user perceives the utility of service being sold to him/her, and about the link characteristics. The optimization problem itself and the assumptions to it are presented and discussed in the following sections.

3.1 The Optimization Problem

The proposed algorithm maximizes total utility, i.e. the sum of user utilities $u_i(x_i, \tau)$ as in (1), but here user i-th associates the utility with both the bandwidth x_i being allocated to him, and the observed delay τ. The solution is constrained by link characteristics: observed delay τ depends on link load, which is the sum of bandwidth allocated to users. So the optimization problem is as follows:

$$\max_{\substack{m_i \le x_i \le M_i,\ i=1,\dots,n,\\ 0 \le \tau \le \tau_{\max}}} \sum_{i=1}^{n} u_i(x_i, \tau), \qquad \text{subject to} \quad h\left(\sum_{i=1}^{n} x_i\right) = \tau \tag{2}$$

Function $h(\cdot)$ in (2) is the link characteristics and is unknown both to users and to the coordination algorithm. Its properties are discussed in section 3.3, while in section 3.2 required properties of utility functions are discussed and a class of utility functions is proposed.

Unlike in schema by Low and Lapsley, the dual of this problem is not easily decomposable due to nonlinearity of the constraint. Instead another approach to decomposition of such problem is proposed in Sec. 4. Here the general idea is presented to show which properties of functions $h(\cdot)$ and $u_i(\cdot)$ are required.

For a fixed value of τ the problem reduces to the following:

$$\max_{m_i \le x_i \le M_i,\ i=1,\dots,n} \sum_{i=1}^{n} u_i^{\tau}(x_i), \qquad \text{subject to} \quad \sum_{i=1}^{n} x_i = c^{\tau}, \qquad \text{where} \quad c^{\tau} = h^{-1}(\tau) \tag{3}$$

This is almost exactly the link bandwidth allocation problem (1). The only difference is that we have equality constraint here. The bandwidth pricing algorithm proposed in [11] can be applied successfully, but here the price p can theoretically become negative.

Such observations lead to an idea of a two-phase algorithm. In the inner loop (first phase), a problem is solved for fixed value of τ to obtain bandwidth allocations for users. In the outer loop (second phase) both the optimal value of τ and the approximation $\hat{h}(\cdot)$ of the unknown function $h(\cdot)$ are sought, the latter being based on link state observations. This way acceptable bandwidth allocations are calculated is very fast, making it possible to handle users connecting or disconnecting continuously. The delay is being adjusted to maximize users' satisfaction and to react to current traffic characteristics and the resulting link state.

3.2 User Utility Model

In order to guarantee the bandwidth allocation problem in the inner loop to be solvable using Low and Lapsley algorithm, the utility function need to be strictly concave, monotonic (increasing for x_i and decreasing for τ) and continuously differentiable. It is proposed to use the following class of utility functions:

$$u:\langle 0;c\rangle \times \langle 0;\tau_{\max}\rangle \to R$$
$$u(x,\tau) = \ln\left(x^{0.5-a} \cdot (\tau_{\max} - \tau)^{0.5+a} + 1\right) \quad (4)$$
$$a \in (-0.5;0.5)$$

Functions of class (4) can model various user preferences: those favoring strict QoS guarantees (high values of parameter a) and those more liberal (low values of a). Exemplary contour plots of the two different utility functions, drawn at the same levels, are presented in Fig. 1. The kind of utility functions proposed maintains nice properties of monotonicity, concavity and continuous differentiability postulated in [11], being prerequisites for problem decomposition. But, despite its formal properties, $u_i(x_i,\tau)$ must also reasonably reflect the true user valuation for the service.

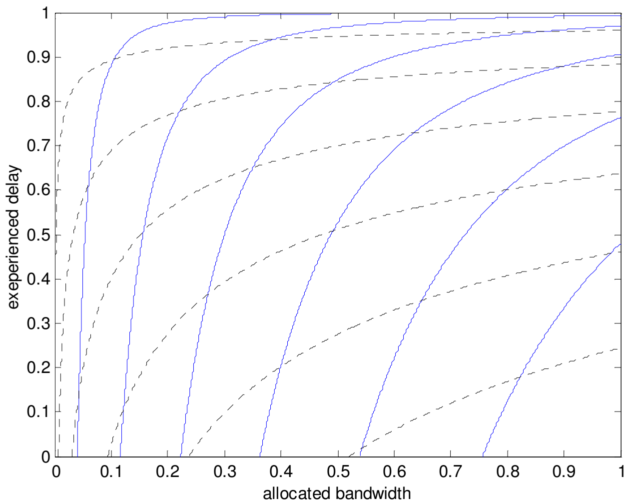

Fig. 1. Contour plots of various user utility functions of the proposed class (solid line for a=-0.2, dashed line for a=0.2).

Increasing user valuation for increasing bandwidth remains something commonly recognized, also for real-time applications: the currently used codecs can be set to produce constant bitrate streams at almost any rate of the choice, usually compromising on image blur and pixelization at lower rates. Therefore, the overall user valuation for bandwidth need not be stepwise increasing at all.

As far as the transmission delay is concerned, there are a number of references on the impact of delay on the overall quality of the service (and, consequently, on the service valuation by the user). In the opinion of experts [3], the delay actually impacts the perceived QoS more than packet loss. This is because currently used applications can deal pretty well with lost packets [3], and because such effects are more tolerable by users (cf. tolerance for cracks vs. response delay in a phone call). The values of transmission delay have been categorized by ITU-T [6] roughly in three intervals, with boundaries at 100 and 400 milliseconds, but those values are rather fuzzy when individual preferences are considered. Indeed, the averaged valuation for delay is decreasing steadily [12], being similar to (4). The division lines for the delay are drawn sometimes arbitrarily, with common value for perceptible and disturbing delay at 0.3 sec, but some authors suggest setting it at 0.25 sec [5].

An interesting delay-dependent utility model, already mentioned, is presented in [4]. The decrease of the utility is linearly dependent on the experienced delay, times the currently allocated bandwidth – and referenced to as a congestion cost. The reasoning behind such model is that in fact the benefits lost on account of delay growth are the bigger the more bandwidth one is using, as they affect every single packet being sent (and paid for) at a specified rate. The utility model proposed in this paper is different, and has its justification, too. We believe that a small, initial increase of delay, does not count as much as further degradation, and so our utility function remains non-linear. Moreover, in the model presented in [4] the delay is more intolerable when the allocated bandwidth is high. This does not necessarily have to be true: a customer pays for application use once, and perceives bandwidth and delay jointly. There exists no reason that he will give up using the application (which, presumably, happens when his utility reaches zero value) at higher transmission rate, while he prefers to use it at lower transmission rates – which happens in model given in [4].

Concluding, we believe that utility function proposed here can accurately model individual user valuation for a service like audio and videoconferencing. It can also be applied to other networking applications. Moreover, such utility can model preferences of business customers (e.g. small operators), competing for bandwidth – especially when those preferences result from aggregation of individual preferences of customer's clients placed in one market segment.

3.3 Delay Model

As it was mentioned above, the problem is going to be solved by repeated adjustments of the assumed delay in the network. That delay value, and the resulting total link utilization, will parameterize the inner problem of price setting with aim to satisfy the resource constraint. The delay adjustments done in the outer loop will be based on hints obtained from customers themselves; therefore the necessary condition for the optimization problem is that the objective function must not have local maxima. Note that in case of utility functions of type (4) the utility maximization problem with allocated total bandwidth and the observed delay treated as two independent variables has a concave and monotonic objective function. If we now consider the link delay model as an extra equality constraint, the resulting one-dimensional problem can be solvable by a sort of gradient algorithm.

It can be observed that in our case, a simple linear link delay model would imply a concave optimization problem. Moreover, as discussed below, the real link characteristics is usually convex and strictly increasing w.r.t. the link load. Therefore, both any contour set of the objective function and the link characteristics are convex and, consequently, we can find exactly one such contour that is tangent to the link characteristics at exactly one point. This is the solution of our problem – the only one, and no other local solution exists. We will show that such convex link delay characteristics are generally found in the real world.

The question about our algorithm applicability is if the observed delay really depends on link utilization in the postulated, convex way. Here appears a question what the term delay means. Most of research papers deal with the average delay, calculated over sufficiently long period. Such understanding is also shared in ITU-T recommendations, cited above.

Poisson process often constitutes the reference load-delay model of a link. It is a stochastic process used to model random events in time that occur independently. The number of events occurring in any given period of time has a Poisson distribution p.d.f. Poisson process is said to be memoryless – the number of events occurring in any bounded interval of time is independent of number of events that occurred before. The simplest form of Poisson process is a homogeneous Poisson process, where intensity or rate of the process (usually denoted by λ) is constant over time. In such case probability of k events occurring in any given time interval t is equal to $e^{-\lambda t}(\lambda t)^k/k!$ and the average number of events is equal to λt.

If arrivals of packets on the link are described by Poisson process with intensity λ (so that the average normalized link load equals λ) and packet processing time is t_0, then average delay on the link is equal to $t_0/(c- \lambda t)$.

Poisson process has large variation, and the average value can be reliably measured only after very long observation. Usually Poisson process is used to describe Internet traffic arriving from many sources (except for unusual situations like denial-of-service attacks), as well as for many other phenomena in physics, telecommunication and in general for study of queuing systems.

In order to capture phenomena like traffic burstiness or self-similarity, a number of traffic models have been proposed. Examination of the output from such models shows that the average transmission delay grows with link utilization, and with the degree of traffic self-similarity, and is generally bigger than predicted by Poisson process [14]. Moreover, those advanced models still underestimate the actual delay, calculated from link traffic traces. In any case, both the observed and modeled delay graphs are convex w.r.t. the traffic load.

Rather than relying on simple means, one can require that the delay of a specified portion of packets should not exceed some maximum delay. Alternatively, the probability of incurring some target delay by a packet should be at certain level. In such case effective bandwidth theory determines how the resources, traffic and packet loss or delay are related. Particularly, requirements for buffer sizes in order to maintain the target packet loss probability, as given in [13], are of use for us. After a closer look at how the required buffer size is influenced by the link load, it turns out that such function is strictly increasing in its whole domain, but it is convex only when the link utilization factor is greater than some 0.1. Since recommended buffer size is related to the packet queuing delay, we can use it as an estimate of QoS understood as the maximum delay experienced with specified probability. Therefore, we can expect our initial assumptions on link characteristics convexity to hold in the bigger part of the optimization domain.

Later simulation verification of the theoretical formulae from [13] are given in [2]. The simulations carried out for a number of video streams multiplexed on an ATM link show that the theoretical formulae tend to be over-optimistic. One can expect that video streaming in IP networks is yet more difficult to be guaranteed certain level of quality. The examination of this topic is considered to lie out of the scope of this paper.

4. ALLOCATION ALGORITHM

It was already stated that proposed algorithm consists of two loops: inner loop for calculating bandwidth allocation for given value of delay τ, and outer loop for finding optimal value of τ and for continuous improvement of approximation of link characteristic $h(\cdot)$.

The inner problem is solved using Low and Lapsley algorithm. Users have to agree on the delay imposed by the coordinator/provider. They have no other choice because the coordinator adjusts price until the allocated bandwidth reaches the assumed total and results in assumed delay value (provided link characteristic is known or approximated well enough).

It is worth to note that this phase does not require any observations of link state, as it relies only on the mathematical link model possessed by the coordinator. Therefore it can be done as fast as data between clients and coordinator can be exchanged, without involving network reconfiguring or monitoring. Moreover, after a small change of proposed delay usually only one or two iterations are enough to tune bandwidth allocations.

In outer loop the delay value τ must be modified to maximize the total utility, which must coincide with the users' individual maximization of their profit. An assumption was made already that users' utility functions are private so the algorithm does not know their shapes. However, some feedback from users is necessary in order to infer about the direction of desirable delay adjustment. It is proposed that users periodically (every time the inner problem is to be solved) send their budget q_i for quality improvement (delay decrease), and coordinator modifies τ accordingly.

To devise a formula for delay change in outer loop, some observations must be made: inner loop has just finished and all clients have chosen optimal bandwidth for a given price p and delay τ. So, for every i, $\partial u_i(x,\tau)/\partial x_i - p = 0$. According to Kuhn-Tucker theorem (x,τ) is optimal if following statements hold:

$$\frac{\partial}{\partial \tau}\left(\sum_{i=1}^{n} u_i(x_i,\tau) - \lambda\left(h\left(\sum_{i=1}^{n} x_i\right) - \tau\right)\right) = 0 \quad (5)$$

$$\frac{\partial}{\partial x_j}\left(\sum_{i=1}^{n} u_i(x_i,\tau) - \lambda\left(h\left(\sum_{i=1}^{n} x_i\right) - \tau\right)\right) = 0,\ j = 1,\ldots,n \quad (6)$$

First, $\partial u_i(x,\tau)/\partial x_i = p$ can be substituted to (6). Then λ can be calculated from the (5) and also substituted:

$$p - \lambda \cdot \frac{\partial h}{\partial x_j}\left(\sum_{i=1}^{n} x_i\right) = 0 \quad (7)$$

$$\lambda = -\sum_{i=1}^{n} \frac{\partial u_i}{\partial \tau}(x_i,\tau) \quad (8)$$

$$p + \sum_{i=1}^{n} \frac{\partial u_i}{\partial \tau}(x_i,\tau) \cdot \frac{\partial h}{\partial x}\left(\sum_{i=1}^{n} x_i\right) = 0 \quad (9)$$

So solving the optimization problem is equivalent to solving (9). If the expression on the left hand side of (9) is positive, the current value of τ should be increased in order to approach the solution; otherwise it should be decreased.

Note that $\partial u_i(x_i,\tau)/\partial \tau$ is the only information about private user utility function that is needed to find optimal delay τ. So in outer loop users should set their budgets to $q_i = \partial u_i(x_i,\tau)/\partial \tau$. Then proposed delay τ should be changed proportionally to $p - Q \cdot \sum_{i=1}^{n} q_i$, where $Q = \partial h\left(\sum_{i=1}^{n} x_i\right) \Big/ \partial x$.

It was previously stated that the algorithm should not make too many assumptions about way traffic influences delay – so function $h(\cdot)$ should also be considered unknown. It is proposed to use any approximation $\hat{h}(\cdot)$ that can be gradually improved using link observations in every iteration of the outer loop. Simulations shown that even the simplest curve fitting algorithm performs well enough, but further research is needed to choose the best method.

The whole algorithm can be written in pseudo code as follows:

Step 1. Choose initial approximation $\hat{h}(\cdot)$, initial delay proposition τ and initial price p.
Step 2. Calculate $c^{\tau}=\hat{h}^{-1}(\tau)$.
Step 3. Communicate τ and p and let users calculate their optimal bandwidth demand x_i.
Step 4. Calculate $\Delta p=(\Sigma x_i - c^{\tau})$
Step 5. If Δp=0 go to Step 8.
Step 6. Adjust price, $p:=p+\Delta p$.
Step 7. Go to Step 3 {inner loop}.
Step 8. Users calculate and communicate their budgets for delay decrease q_i.
Step 9. Observe link load and delay for some period of time.
Step 10. Improve approximation $\hat{h}(\cdot)$ using pair observations from Step 9.
Step 11. Calculate new delay proposition τ based on p, Σq_i and $\hat{h}(\cdot)$.
Step 12. Go to Step 2. {outer loop}

Table 1. Comparison of results obtained by the algorithm with reference values for different preferences of users A and B.

Test number	Source preferences (parameter a)		Bandwidth allocation				Delay	
			Algorithm (ns-2)		Reference (Matlab)		Algorithm	Reference
	A	B	A	B	A	B		
1	-0.1	0.1	0.2642	0.4210	0.2642	0.4209	0.3176	0.3175
2	-0.1	-0.1	0.3186	0.3186	0.3184	0.3184	0.2755	0.2754
3	-0.4	0.0	0.0977	0.4807	0.0976	0.4804	0.2371	0.2370
4	0.0	0.4	0.2249	0.5487	0.2249	0.5488	0.4418	0.4419
5	-0.2	0.3	0.1716	0.5418	0.1715	0.5417	0.3488	0.3487

5. SIMULATION RESULTS

Simulations were conducted in two environments: Matlab (with known link characteristics) and in ns-2. Matlab simulations were to check if solution found by the algorithm is optimal, while ns-2 tests should check algorithm stability in situation close to real network usage.

Matlab tests were conducted as follows: there were two sources connected to one link of capacity 1 and packet processing time 0.1. It was assumed that traffic is described by Poisson process. Sources' preferences (described by parameter a) were different in each test. Comparison of algorithm results with reference solution is presented in Table 1.

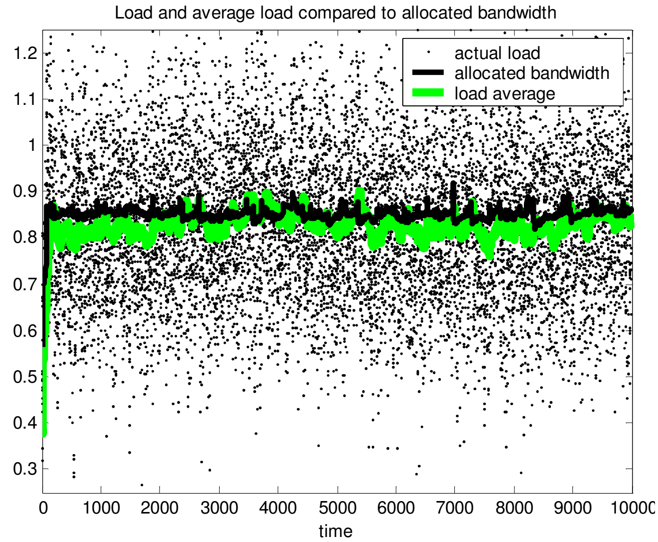

Fig. 2. Allocated bandwidth vs. link utilization simulated in ns-2.

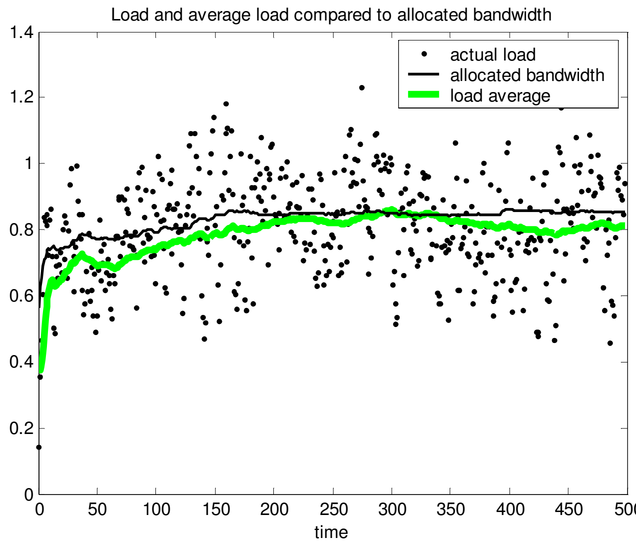

Fig. 3. Bandwidth allocation – startup phase of the algorithm.

In Matlab environment it was only possible to simulate algorithm behavior while assuming some average values of traffic bandwidth and delay, so set of tests was performed in ns-2 to prove that the algorithm is stable in more real situation, where temporary link state is varying and mean values can be observed in long term only. Two Poisson sources were introduced to a network consisting of two nodes and a link between them. As the algorithm was changing bandwidth allocations, sources' rates were modified accordingly. Traffic and delay were observed in constant time intervals and measured values were used to improve link characteristic approximation. Link load, calculated mean values and algorithm allocations are presented in Fig. 2. One can see that allocation is quite stable though link load is very variable. Also, the algorithm quickly reacts to startup conditions that are far from the problem solution, as can be seen in Fig. 3 where a close-up of Fig. 2 is presented.

Next, the delay chart is presented in Fig. 4. One may notice that the actual averaged delay is generally greater than planned, or predicted, delay. Such prediction bias is probably due to imperfect link delay model. This drawback must be considered a serious one as the individual customers' decisions are based on trust to the provider in its declared QoS. It turns out that delay prediction model must be more conservative, especially if it is to work in real network where Poisson model turns out to be very far from perfect.

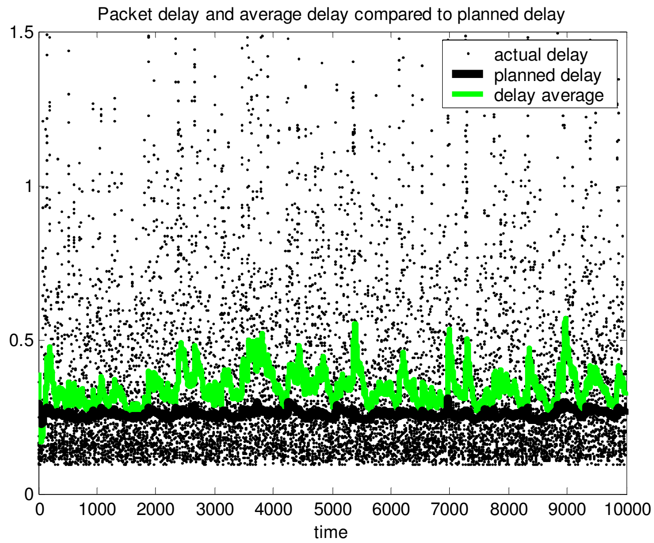

Fig. 4. Predicted and real link average link delay.

6. CONCLUSIONS

The research reported in this paper shows that it is possible to maximize social utility of a transmission link also in case when the link users have various preferences for QoS, and do not want to disclose them altogether. The bandwidth allocation algorithm is based on the well-known and socially acceptable approach of a single linear tariff for all users. The only requirement that makes it possible to look for the optimal link load (and delay) is that users should report their actual price for quality improvement.

Simulations have proven stable and efficient operation of the algorithm, which is partly due to bandwidth allocation procedure based on the current estimate of link characteristics. However, this quick open-loop mode is periodically abandoned with aim to feed the algorithm with real-life data collected directly from the link.

We believe that after necessary improvements of the link model the algorithm will be able to execute adaptive control over the real network. Apart from that, the future work will focus on the model extension to multiple link case, and on harnessing the existing QoS technologies, as DiffServ, to support operation of this algorithm in real network.

ACKNOWLEDGMENTS
The work reported in this paper has been partially supported by Polish Ministry of Science and Higher Education research grant PBZ-MNiSW-02/II/2007.

REFERENCE LIST

[1] Alpcan, T. and Başar, T. 2005. "Utility-Based Congestion Control Scheme for Internet-Style Networks with Delay", IEEE/ACM Transactions on Networking, 13, no. 6 (December): 1261-1274.

[2] Bashforth, B.N., and Williamson, C.L. 1998. "Statistical Multiplexing of Self-Similar Video Streams: Simulation Study and Performance Results". In Proceedings of the Sixth International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems, MASCOTS. 119-126.

[3] Claypool, M., and Kinicki, R., and Kumar, A. 2005. "Traffic Sensitive Active Queue Management." In Proceedings of 24th Annual Joint Conference of the IEEE Computer and Communications Societies, INFOCOM 2005: 2764-2769.

[4] Courcoubetis, C., and Weber, R. 2003. *Pricing Communication Networks: Economics, Technology and Modelling*. John Wiley & Sons.

[5] Haßlinger, G. 2004. "Implications of Traffic Characteristics on Quality of Service in Broadband Multiservice Networks." In Proceedings of the 30th EUROMICRO Conference.

[6] International Telecommunication Union - Telecommunication Standardization Sector (ITU-T), 2006. Network performance objectives for IP-based services. Recommendation Y.1541.

[7] Jaskóła, P., and Malinowski, K. 2004. „Two Methods of Optimal Bandwidth Allocation in TCP/IP Networks With QoS Differentiation". In Proceedings of the 2004 Summer Simulation Multiconference, SPECTS 2004 (San Jose, CA): 373–378.

[8] Karpowicz, M., and Malinowski, K. 2007. "Characterization of scalar strategy mechanisms for efficient resource allocation: Supply side case." In Proceedings of 46th IEEE Conference on Decision and Control, CDC 2007 (New Orleans, LA): 150-155.

[9] Kelly, F.P., and Mauloo, A., and Tan, D. 1998. "Rate Control for Communication Networks: Shadow Prices, Proportional Fairness and Stability, Journal of Operations Research Society, 49, no. 3 (March): 237-252.

[10] Kozakiewicz, A. 2008. "Effective Load Theory for Pricing and QoS Control of Computer Networks." Ph.D. dissertation in Faculty of Electronics and Computer Science, Warsaw University of Technology.

[11] Low, S.H., and Lapsley, D.E. 1999. "Optimization Flow Control, I: Basic Algorithm and Convergence", IEEE/ACM Transactions on Networking, 7, no. 6 (December): 861-874.

[12] Mustill, D. 2006. "Network Performance Objectives for IP-Based Services: Challenges & Opportunities." Presentation on Global Seminar on Quality of Service and Customer Protection, Geneva, Switzerland. ITU.

[13] Norros, I. 1995. "On the Use of Fractional Brownian Motion the theory of Connectionless Networks", IEEE Journal on Selected Areas in Communications, 13, no. 6 (August): 953-962.

[14] Shu, Y., and Xue, F., and Zhigang. J., and Yang, O. 1998. "The Impact of Self-similar Traffic on Network Delay." In Proceedings of IEEE Canadian Conference on Electrical and Computer Engineering: 349-352.