International Journal of Computing and Network Technology

http://dx.doi.org/10.12785/ijcnt/010204

Resource Allocation with Delay QoS

Hengky Susanto[1] and ByungGuk Kim[2]

Department of Computer Science, University of Massachusetts - Lowell, MA, Lowell, USA
[1]Email Address: hsusanto@cs.uml.edu
[2]Email Address: kim@cs.uml.edu

Received: 14 Aug. 2013, Revised: 18 Aug. 2013; Accepted: 23 Aug. 2013
Published online: 1 Sep. 2013

Abstract: This paper primarily addresses how available bandwidth should be optimally distributed among competing streams of elastic traffic like TCP traffic while taking Quality-of-Service (QoS) and delay into consideration. Network Utility Maximization (NUM) in [1], a congestion control algorithm, allows users to set for an optimum network-wide rate allocation through their utility. By incorporating delay into utility function, users can accommodate for QoS requirements.

Keywords: networks, congestion control, resource management, optimization methods, QoS.

I. INTRODUCTION

With the increasing demand for bandwidth along with the increasing size of data network transmission, the network becomes overburdened and user may experience connection quality degradation. The mismanagement of bandwidth allocation may lead to bottleneck where the amount of data that is transmitted into the network exceeds the capacity. If the demands exceed the capacity, performance is generally poor and unpredictable. Thus, an appropriate model for bandwidth allocation becomes an important task in assuring high network performance. Network bandwidth allocation was formulated as a Network Utility Maximization (NUM) problem by F. Kelly [1][2]. The NUM formulation attempts to maximize the aggregate utility of users receiving bandwidth subject to limits on the link capacity,

$$maximize \sum_{s \in S} U_s(x_s)$$

$$s.t.\ Ax \le Cm\ \ Hy = x$$

$$over\ x, y \ge \bar{0}$$

Here, C denotes a set of capacity of link l, $for\ \ l \in L$, where L denotes a set of links in the network and S, a set of users accessing the network. The matrix A has the routing information that link l is associated with route r and matrix R has the path of user s, such that $A=(A_{lr}, l \in L, r \in R)$

, where $A_{lr}=1$ if $l \in r$, and $A_{lr}=0$, otherwise. Let $H_{sr}=1$ if path r is associated with user, and $H_{sr}=0$, otherwise resulting in the matrix $H=(H_{sr}, s \in S, r \in R)$. Variable y is a set of flow traverse over router. In addition, the utility function is assumed to be non-decreasing smooth function, strictly concave, and differentiable in $x>0$ These conditions are necessary for convex optimization [2]. Kelly has demonstrated that network traffic flows can be regulated in a *proportionally fair* manner with a distributed approach [1].

Kelly's framework was extended to various issues. NUM was used to model network protocols by "reverse engineering" a given protocol, such as TCP/IP, to provide an "inside look" of the Internet congestion and to obtain fair bandwidth allocation [4][5][6]. In [7][8], delay function influences user's utility and bandwidth allocation scheme were discussed. Furthermore, a more general utility function that considered NUM and delay in VoIP was discussed in [9] by considering the queuing theory in order to influence the bandwidth allocation by adjusting the delay requirement. This resulted in degrading the performance of lower-priority traffic and the algorithm is VoIP-specific. Delay functions were incorporated into NUM in [11][12][13] by taking the delay function as the network cost which reduced user's utility. Delay functions in [14][15] were formulated as a ratio between the bandwidth capacity and the buffer occupancy. In this paper, we focus on how to accommodate diverse elastic applications in a network where a mix of traffic may have different requirements for bandwidth and Quality of Service (QoS). As QoS, we consider packet delays and we propose a delay utility function based on *M/M/1* queuing results [10].

II. DELAY UTILITY FUNCTION

Let QoS be expressed by the average delay through the network. When a user has the bandwidth x allocated in a link, it can be considered equivalent to the transmission rate for the user. The *M/M/1* based delay function [10], $d(x)$, is defined as the average delay (including transmission time) in a link:

$$d(x_s)=\frac{\alpha}{x_s(x_s-\alpha)}+\frac{1}{x}$$

$$=\frac{1}{x_s-\alpha}, \text{ for } x>\alpha\geq 0, \tag{1}$$

where α denotes the arrival rate at a link. In this context, x_s can be interpreted as processing rate of user s. Thus, the delay of the entire path is $\sum_{l\in r} d_l(x_s)$, where route r is a set of link $l \in L$ that connects source and sink. The delay utility function U_{QoS} is then formulated so that it increases or decreases according to the delay. This is similar to the idea in [3], where the degree of user's satisfaction over shorter delay diminishes as the traffic gets smoother. We thus define delay utility function in a single link as follows.

$$U_{QoS}(x)=m_q \log\left(\frac{1}{d(x)}\right)$$

$$=m_q \log(x-\alpha), \tag{2}$$

where m_q is the user's willingness to pay for quality. The *delay utility function* of a traffic stream traversing a path r is given by

$$U_{QoS}(x) = m_q \log(\sum_{l \in r} x_l - \alpha_l) \tag{3}$$

The bandwidth allocation problem with the delay QoS is formulated as the following optimization problem.

$$\begin{aligned} &maximize \sum_{s \in S} U_s(x_s) \\ &s.t.\ Hy = x,\ Ay \le C, \\ &over\ x, y \ge 0, \end{aligned} \tag{4}$$

where $U_s(x_s) = U_{bw}^s(x_s) + U_{QoS}^s(x_s)$, user utility function given allocated bandwidth x_s is

$$U_{bw}^s(x_s) = m_{x,s} \log(x_s), \tag{5}$$

And $m_{x,s}$ is user s is willingness to spend on bandwidth x_s as it is proposed by Kelly in [1][2]. The other variables are identical to those in (2). Furthermore, the allocated bandwidth x_s must be bounded in the function $U_{Qos}(x_s)$ because delay $d(x_s) = \frac{1}{x_s - \alpha_s} > 0$ must be satisfied. Otherwise, the queue length will grow exponentially, which leads to further performance degradation. Thus, $U_{Qos}(x_s)$ is modified as follows.

$$U_{Qos}(x_s) = \begin{cases} m_q \log(\sum_{l \in r} x_s - \alpha_s) & , \quad (x_s - \alpha_s) > 0 \\ 0 & \quad Otherwise \end{cases}$$

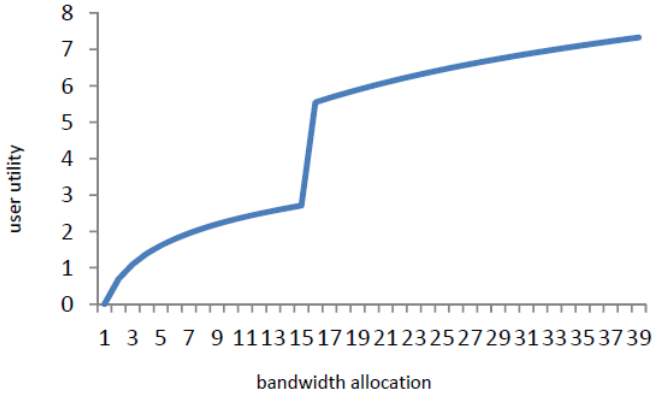

Figure 1. Non-convex utility function

However, without providing constraint $x_s \geq \alpha_s$ to $U_s(x_s)$, it may lead to non-convex situation because function $U^s_{QoS}(x_s)$ is not continuously concave and that cause function $U_s(x_s)$ to be non-convex as shown in figure 1. Furthermore, notice the derivative of utility function $U_{QoS}(x_s)$, when $U_{QoS}(x_s) = 0$ and $x_s \leq x_s^{\min}$, is $\frac{dU^s_{QoS}(x_s)}{dx_s} = U'_{QoS}(x_s) = 0$. Conversely, $U'_{QoS}(x_s) > 0$ when $x_s > x_s^{\min}$. So, when $x_s > x_s^{\min}$,

$$\int_0^{x_s} U_{QoS}(y)dy = \int_0^{x_s^{\min}} U_{QoS}(y)dy + \int_{x_s^{\min}}^{x_s} U_{QoS}(y)dy$$

$$= \int_{x_s^{\min}}^{x_s} U_{QoS}(y)dy > 0.$$

As we observed, when $x_s > x_s^{\min}$, utility function $U_{QoS}(x_s)$ is an increasing and strictly concave function, which is convex. So, that allows the addition of two convex functions, $U^s_{bw}(x_s)$ and $U^s_{QoS}(x_s)$, is also convex. Therefore, for $U^s_{QoS}(x_s)$ to be convex, it must satisfy $x_s > x_s^{\min}$.

For that reason, to preserve convexity and prevent from becoming into non-convex problem, the problem of utility maximization is reformulated as follows.

$$maximize \sum_{s \in S} U_s(x_s)$$

$$A_x \leq C,$$

$$x - x^{\min} \geq 0,$$

$$\text{over } (x^{\min} \geq 0).$$

Therefore, in this thesis, we assume the utility functions are continuous and twice differentiable on $(0, +\infty)$ and satisfy the following properties:

1. $U_{bw}(x_s), U_{QoS}(x_s) \geq 0, \forall x_s$.
2. $\forall x_s$, $0 \leq x_s^{\min} \leq x_s$ and $U_{bw}(0)$, $U_{QoS}(0) = 0$, $\forall x_s, s$.
3. $U_{bw}(x_s)$ and $U_{QoS}(x_s)$ are twice differentiable.
4. $U_{bw}(x_s)$ and $U_{QoS}(x_s)$ are concave.

5. $\frac{dU_{bw}(x_s)}{dx_s}, \frac{dU_{QoS}(x_s)}{dx_s} < \infty$, for all $0 \le x_s \le c$.

6. $\lim_{x_s \to 0} \frac{dU_{bw}(x_s)}{dx_s}, \lim_{x_s \to 0} \frac{dU_{QoS}(x_s)}{dx_s} < \infty, \forall_s$.

7. $\sum_{s \in S(l)} x_s^{\min} \le C_l$.

8. $\forall x_s, \sum_{s \in S(l)} x_s^{\min} \le \sum_{s \in S(l)} x_s \le C_l$, for $x_s^{\min} < x_s$.

The property 7 and 8 assure that $U_{QoS}(x_s)$ convexity and network satisfied of the minimum bandwidth requirement.

III. RATE ALLOCATION

The following step, we solve (4) in the Lagrange form, we can write that

$$L(x, y, z, \lambda, \mu) = \sum_{s \in S} U_s(x_s) - \lambda_s^T (x - H_y) + \mu^T (C - Ay \ - z) \tag{6}$$

where $\lambda = \{\lambda_s, s \in S\}$ are vectors of Lagrange multipliers and $\mu = (\mu_j, j \in J)$ is a vector of slack variables. Since (3) and (5) are convex functions, the Lagrangian form can be directly solved such that

$$\frac{dU_s}{dx_s} = \frac{dU_{bw}^s}{dx_s} + \frac{dU_{QoS}^s}{dx_s},$$

$$= \frac{m_{x,s}}{x_s} + \frac{m_{q,s}}{x_s - \alpha s}. \tag{7}$$

By combining the derivative of (6) and (7), we have

$$\frac{dL}{dx_s} = \frac{m_{x,s}}{x_s} + \frac{m_{q,s}}{x_s - \alpha s} - \lambda_s = 0, \tag{8}$$

which yields $\lambda_s = \frac{x_s m_{x,s} - \alpha_s m_{x,s} + x_s m_{q,s}}{x_s (x_s - \alpha_s)}$, and

$$x_s = \frac{(\lambda_s \alpha_s + m_{x,s} + m_{q,s}) \pm \sqrt{(\lambda_s \alpha_s + m_{q,s} + m_{q,s})^2 - 4\lambda_s \alpha_s m_{x,s}}}{2\lambda_s} \tag{9}$$

The solution of x_s can be obtained from

$$\sum_{s \in l} x_s = C_l, \tag{10}$$

by combining (9) to (10) and solving for λ_s on link l, for all user sharing link *l*, where $l \in L$. It can be interpreted as deciding the network price that users must pay for using link *l*. Once λ_s is obtained, user *s* solves x_s by manipulating of (8),

$$x_s = \frac{m_{x,s} - \frac{\alpha_s}{C} m_x + m_{q,s}}{\lambda_s} + \alpha_s .$$

Additionally, network may also decide the minimum value for m_x and m_q to offset the operation cost. For instance,

$$m_{x,s} = \max(m_{x,s}, \sum_{l \in r_s} \frac{\cos t(l)}{nl}) \tag{11}$$

And

$$m_{q,s} = \max(m_{q,s}, \cos t(q)), \tag{12}$$

where $\cos t(l)$ denotes the operation cost on link *l* [16], n_l is the number of users sharing *l*, $\cos t(q)$ is the operation cost to acquire quality *q*, and for $m_{x,s}$, $m_{q,s}$, $\cos t(l)$, $\cos t(q) \geq 0$. The design of cost function *cost(.)* is beyond the scope of our discussion. Additionally, user may be paying too much when

$$\frac{U_s(x_s)}{m_{x,s} + m_{q,s}} < Threshold .$$

In [1], Kelly has also introduced a concept of fairness.

Definition 1: A vector of rates *x = (xl , l є L)* is *proportionally fair* if it is feasible, that is $x \geq 0$ and $Ax \leq C$, and if for any other feasible vector x*, the aggregate of proportional changes is zero or negative:

$$\sum_{s \in S} \frac{(x_s^* - x_s)}{x_s} \leq 0. \tag{13}$$

However, (13) is not sufficient when $x_s^* \leq \lambda$ because delay function $d(x^*) = \frac{1}{x^* - \alpha} \leq 0$. Thus,

$$x^* = \begin{cases} x^*, & \text{if } x^* - \alpha > 0 \\ x, & \text{otherwise} \end{cases} \tag{14}$$

Corollary 1: Condition (14) satisfies (13).

Proof: There is other feasible vector *x** that is *proportionally fair* and $x_s^* \leq \alpha_s$. But by condition in (14), $x_s^* = x_s$ then $\sum_{s \in S} \frac{(x_s^* - x_s)}{x_s} = \sum_{s \in S} \frac{(x_s - x_s)}{x_s} = 0$. Otherwise $\sum_{s \in S} \frac{(x_s^* - x_s)}{x_s} \leq 0$. Thus satisfies (13).

IV. DISCUSSION

4.1 The Impact of User Willingness to Pay

Consider a simple configuration with two nodes connected by a single link *l* with capacity *C*=100. The link *l* is shared by flows *a* and *b*. In order to investigate the effect of m_q, we assume that $m_{a,x} = m_{x,b} = 10$ and $m_{q,a} = m_{q,b} = 10$, initially. When $\alpha_\alpha = 60$ and $\alpha_b = 1$, the initial bandwidth allocation equally divides the excess capacity between the two flows ($\frac{C - \alpha_a - \alpha_b}{2} = 19.5$) such that x_a = 79.5 and x_b = 20.5.

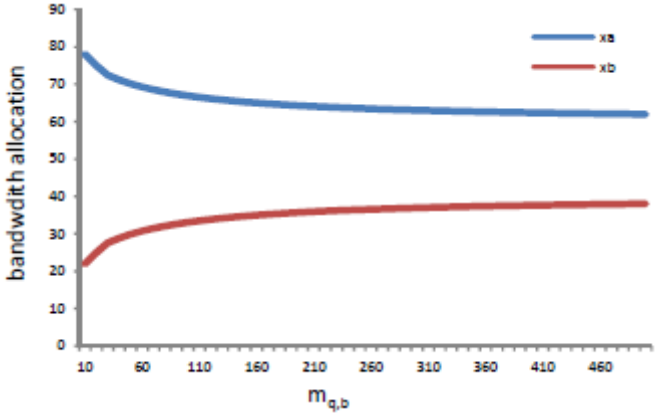

Figure 2. the relationship of bandwidth distribution between user a and b.

In Fig. 2, we plot bandwidths as $m_{q,b}$ (the willingness to pay for QoS) increases for flow *b*. The figure shows that bandwidth for *b* increases (for *a* decreases) and converges.

4.2 Parking Lot Configuration

As a potentially congested example, a parking lot configuration with four nodes is considered.

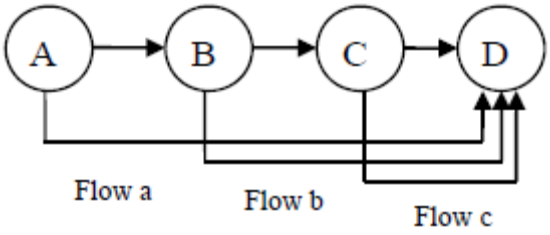

Figure 3. Single Bottleneck.

Three flows *a*, *b*, and *c* share links as depicted in the fig. 3. they have the same values for m_x and m_q , as in Table 1, the bandwidth is identical $(x_a = x_b = x_c = 33.333)$ as they equally share the link C_{CD}. We assume that link capacities are identical in three links with $C_{AB} = C_{BC} = C_{CD} = 100.$ For simplicity, $\lambda_s = \max(\{\lambda_{s(l)}\} \mid l \in r_s)$.

Table 1. (Case 1)

	M_x	M_Q	α
User a,b,c	10	5	1

Suppose that flow *c* increases the willingness to pay from $m_{q,c} = 5$ to 50 and 100(it remains the same in other flows). The resulting bandwidths in three flows are listed in Table flow *c* now receives more bandwidth.

Table 2. (Case 2)

	x_a	x_b	x_c
$m_{q,c} = 5$	33.33	33.33	33.33
$m_{q,c} = 50$	20.025	20.025	59.37
$m_{q,c} = 100$	13.46	13.46	72.1

This demonstrates that QoS specification in terms of the willingness to pay for QoS can be used as a parameter when a flow requests bandwidth.

4.3 Diverse Users' Demands

Consider the next example of 6 flows competing for the core of network. The capacity on every link is

assumed to be identical with $C_l = 100$. Similar to the previous simulation, $\lambda_s = \max(\{\lambda_{s(l)}\} | l \in r_s)$.

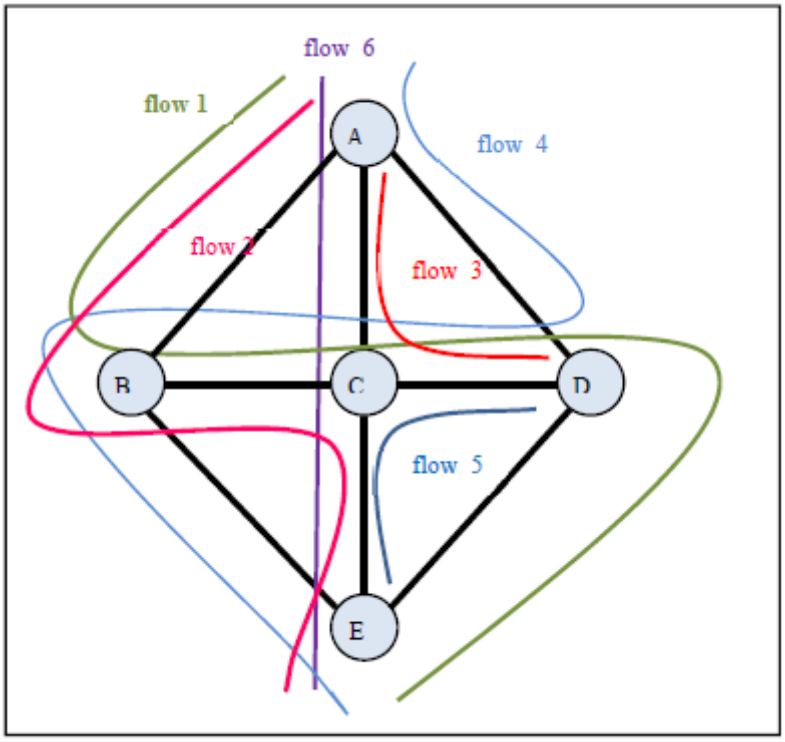

Figure 4. Bandwidth Allocation in a Core Network.

The configuration of the network is illustrated in in Fig. 4, where m_x , m_q , and α , and the result from bandwidth allocation *x* for each flow and its delay is listed in table 3. The delay of each flow $d(x)$ is the summation of delay occurrence on each link along the path which the flow traverses.

Table 3.

	flow 1	flow 2	flow 3	flow 4	flow 5	flow 6
m_x	10	10	10	10	100	10
m_q	10	10	20	50	0	50
α	20	20	10	10	0	20
x	23.51	27.51	17.45	27.31	30.72	39.48
d(x)	1.12	0.39	0.26	0.23	-	0.12

This example illustrates the usage of willingness to pay and user's demand for bandwidth to influence the flow's delay. For example, flow 4 and 6 are example of flows with high QoS demand, where flow 1 and 2 are flows with lower QoS demand. Additionally, flow can also only require bandwidth without considering the quality like flow 5. Moreover, flow 5 is a special case of unusually high demand for bandwidth allocation without QoS guarantee. Notice that flow 3 can reduce its delay if flow 3 traverses over link *AD* instead of *AC*

and *CD* because less bottleneck in *AD* and less number of hops from source to sink. For instance, flow 3's delay can be reduced from $d(x_s)$ = 0.26 to 0.039 and achieves higher utility if flow 3 traverses over AD.

CONCLUSION

In this paper, we address the impact of incorporating QoS to utility function and present a pricing schemewhich takes user's requirement for QoS into consideration. On the other hand, our model does not consider propagation delay because it is assumed to be constant. In addition, queuing model provides long term average value and input data is measured over an extended period of time. Furthermore, the proposal model only supports elastic traffic like email, FTP, HTTPs, and others like these. However, we need to consider real time traffic, which is a non-convex, and the extension of the model to include non-convex traffic will be the topic of further studies.

References

[1] F. P. Kelly, Charging and rate control for elastic traffic, *European Transaction on Telecommunication*, **8**(1), (1997), pp. 33-37.

[2] F. P. Kelly, A. Maullo, and D. Tan, Rate control in communication networks: shadow prices, proportional fairness and stability, *Journal of the Operational Research Society*. **49**(3), (1998),pp. 237-252.

[3] S. Shenker,Fundamental Design Issues for the Future Internet, *Journal on Selected Areas in Communications*,**13**(7), (1995), pp.1176-1188.

[4] J. Mo and J Walrand, Fair End-to-End Window-Based Congestion Control, *ACM Transaction*, **8**(5), (1999), pp. 556-567.

[5] T. Lan, D. Kao, M. Chiang, and A. Sabharwal, An Axiomatic Theory of Fairness in Network Resource allocation, Proceedings of *IEEE INFOCOM*, (2010), pp. 1-9.

[6] S. Low and D. Lapsely, Optimization Flow Control, I:Basic Algorithm and Convergence, *ACM Transaction On Networking*, **7**(6), (1999), pp. 861-874.

[7] S. Stidham, Jr., Pricing and congestion management in a network with heterogeneous users, *IEEE Trans. on Auto.Control*, **49**(6), (2004), pp. 976-981.

[8] J. Pongsajapan and S. H. Low, Reverse Engineering TCP/IP-like Networks using Delay-Sensitive Utility Functions, Proceedings of *IEEE INFOCOM, Anchorage, USA*, (2007).

[9] Y. Li, M. Chiang, R. A. Calderband, and S. N. Diggavi, Congestion Control in Networks with Delay Sensitive Traffic, Proceedings of *IEEE GLOBECOM, Washington DC, USA*, (2007), pp.1942-1948.

[10] L. Kleinrock, *Queuing Systems*, (1976), Vol. 1: Theory; Vol. 2: Computer Applications. New York: Wiley.

[11] S. H. Low, A Duality Model of TCP and Queue Management Algorithm,, *IEEE/ACM Trans. Network*, **10**(3), (2003), (2003), pp. 525-536.

[12] D. Mayer and J. Barria, Bandwidth Allocation of a Revenue-Aware Network Utility Maximization, *IEEE Communication Letter*, **11**(7), (2007), pp. 634-636.

[13] M. Saad, A. Leon-Garcia and W. Yu, Optimal Network Rate Allocation under End-to-End Quality-of-Service Requirements, *IEEE Trans. on Net. And Service Management*, **4**(3), (2007), pp. 40-49.

[14] N. J. Keon and G. Anandalingam, Optimal Pricing for Multiple Services in Telecommunication Networks Offering Quality-Of-Service Guarantees, *IEEE/ACM Trans. Networking*, **11**(1), (2003), pp. 66-80.

[15] N. Jin, G. Venkitachalam, and S. Jordan, Dynamic Congestion-based Pricing of Bandwidth and Buffer, *IEEE/ACM Trans.Networking*, **13**(6), (2003), pp. 1233-1246.

[16] A. Couch, N. Wu, and H. Susanto, Toward a cost model for system administration, *Proc. Large Installation System Administration 2005, USENIX Association, San Diego, CA*, (2005), pp. 9-21.