Proceedings of the 5th WSEAS Int. Conf. on Electronics, Hardware, Wireless and Optical Communications, Madrid, Spain, February 15-17, 2006 (pp114-119)

An Enhanced Rate Allocation Scheme for Ephemeral Traffics in High Speed Networks

AMIR ASKARIAN, MOHAMMAD KAZEM MOUSSAVI
Electrical and Computer Engineering Department
University of Tehran
North Kargar St., Tehran
IRAN

Abstract: - We have proposed a high-speed fluid-flow rate allocation scheme. We show that this method is well suited for ephemeral web-like traffic which is proliferating as network speed is increasing. We look at the flow level properties and packet level algorithm. We further discuss the equilibrium properties and show that this algorithm can achieve proportional fairness. The simulation results for both permanent and short-lived flows are promising.

Key-Words: - Congestion control, rate allocation, fairness, TCP, bandwidth sharing, short-lived traffics, high speed networks.

1 Introduction

Congestion control is becoming a more demanding problem than ever as networks grow in capacity. Using conventional congestion control algorithms in these environments will never meet the objectives these algorithms are designed for. An important concern is achieving fair rate allocation. In these high speed environments, since high bandwidth is available for a flow, its life my finish before it reaches the anticipated fair rate, should conventional rate allocation scheme be deployed. The number of these ephemeral flows increases as the network capacity increases, calling for urgent need of congestion control algorithms with high convergence rate suitable for short lived flows.

Different methods of congestion control can be classified in three ways [1]: window based vs. rate based, implicit vs. explicit and hop by hop vs. end to end. In window based scheme, transmitting rate is controlled by error control window, while in the latter a fluid flow notion is used and transmission rate is controlled directly. Here we use fluid flow notion as Kelly [2]. In explicit method, sources solicit explicit congestion information through a probing mechanism or via explicit feedback, while in implicit approach, information about network congestion status should be inferred from implicit feedbacks such as loss probability or queuing delay. The congestion control algorithm may be deployed between adjacent hops in the network or just in the end hosts. TCP congestion control, as the most widely implemented algorithm in the Internet is a window based, implicit and end to end algorithm. Kelly's method is an end to end algorithm but updates the users' rates directly, therefore is rate based though its window based version can also be formulated [3]. This method employs a general concept of feedback information known as price in end hosts. This can be either an implicit or explicit piece of information. Our algorithm falls under the same category as that of Kelly.

Providing fair rate allocation is one of the main concerns of the rate allocation algorithm. Different notions of fairness exist such as max-min criterion [4], which gives priority to sources with least rates, proportional [5] and weighted proportional [2] criteria which tries to maximize the aggregate proportional changes and minimum potential delay fairness [6] which minimizes the aggregate data transfer times for sources and finally in general form we have (Ω, α) fair rate allocation [3]. Kelly has shown that the using logarithmic utility functions for sources, leads to weighted proportional fair rate allocation. As we will see our algorithm will also converge to these rates.

The rest of this paper is organized as follows. In Section 2 we discuss the fundamentals of utility based approach to rate allocation initiated by S. J. Golestani [7], [8] and further developed by F. P. Kelly [2] and S. Low [9]. We also discuss the fairness and equilibrium properties and look at the stability analysis. In Section 3 we present the high speed algorithm and the simulation results. We conclude in Section 4.

2 Network Optimization

Rate allocation in data networks is studied as a distributed optimization problem [2]-[9]. This optimization problem establishes a well defined framework for studying network characteristics, designing new protocols or improving the existent ones. It is known that the currently implemented TCP congestion control also is a special case of this general framework, by which its performance is evaluated and enhancements are proposed. At this section we review rate allocation as the optimization problem.

2.1 Minimum Cost Flow Control

Minimum cost flow control or MCFC proposed by S. J. Golestani [7,8] formulates end to end congestion control as a global optimization problem, based on which a class of algorithms for adjusting end host rates are proposed. This theoretical approach leads to a class of congestion control algorithms that describes the common algorithms like TCP as its special case.

We consider a network consisting of set $\mathbf{L}=\{l=1,\dots,L\}$ of links and set $\mathbf{S}=\{s=1,\dots,S\}$ of sessions. Let x_s denote the average rate of session s traffic and f_l denote the average traffic of link l. The fraction of traffic of session s carried over link l is denoted by φ_{sl}. This defines routing matrix $\Phi=(\varphi_{sl}, s\in\mathbf{S}, l\in\mathbf{L})$. In single path case as is in the Internet, $\varphi_{sl}\in\{0,1\}$. It follows that

$$f_l=\sum_{s=1}^{S}\varphi_{sl}\cdot x_s, \qquad l\in\mathbf{L} \qquad (1)$$

The rates to be allocated to the sessions should satisfy

$$0\le x_s\le x_s^d \qquad (2)$$

where x_s^d is the desired rate of session s.

To formulate the congestion control as a resource allocation and optimization problem, two cost functions are considered in this framework. First there is a user dissatisfaction cost function $e_s(x_s)$, which demonstrates the cost of limiting the rate of session s to x_s. This is considered a decreasing convex function of x_s. There is another cost function $g_l(f_l)$ associated with each link l that goes to infinity as the aggregate flow on link l reaches the link capacity.

Based on the previous assumptions, the network congestion control problem is formulized based on the following optimization problem:

$$\min_{\vec{r}} J(\vec{r})\triangleq\sum_{s=1}^{S}e_s(r_s)+\sum_{l=1}^{L}g_l(f_l) \qquad (3)$$

Subject to:

$$0\le x_s\le x_s^d$$

$$\vec{f}=\Phi\vec{x}$$

Incremental reward function of session s is defined as

$$h_s(x_s)\triangleq-\frac{\partial}{\partial x_s}e_s(x_s) \qquad (4)$$

Since $e_s(x_s)$ is decreasing and convex, $h_s(x_s)$ is positive and decreasing.

Congestion measure of a session s is defined as the increase of network congestion due to a unit increase in x_s, as follows:

$$\gamma_s(\vec{f})\triangleq\frac{\partial}{\partial x_s}\sum_{l=1}^{L}g_l(f_l)=\sum_{l=1}^{L}\varphi_{sl}\cdot g_l'(f_l) \qquad (5)$$

It is always a positive quantity. In the single path routing scenario (5) reduces to:

$$\gamma_s(\vec{f})=\sum_{l\in\rho_s}g_l'(f_l) \qquad (6)$$

where ρ_s is the path used by session s.

It is shown in [8] that using Kuhn-Tucker theory [10] the optimality condition holds for the optimization problem (3) is as follows. Assuming that $g_l(\cdot)$ and $e_s(\cdot)$ have the first and second derivatives satisfying $g_l'>0, g_l''>0, e_s'<0$ and $e_s''>0$, the following is the set of necessary and sufficient conditions for the session rate vector $\vec{x}^*$ to be the solution for the convex optimization problem (3):

$$h_s(x_s^*)\ \begin{cases}\le\gamma_s(\vec{f}^*), & \text{if } x_s^*=0\\ =\gamma_s(\vec{f}^*), & \text{if } 0<x_s^*<x_s^d\\ \ge\gamma_s(\vec{f}^*), & \text{if } x_s^*=x_s^d\end{cases} \qquad (7)$$

For $s\in\mathbf{S}$ and $\vec{f}^*=\Phi\vec{x}^*$.

Accordingly, the iteration of gradient projection algorithm for solving (3) would be:

$$x_s[n+1]=\begin{cases}0, & \text{if } x_s[n]+k\left(h_s(x_s[n])-\gamma_s(\vec{f})\right)\le 0\\ x_s^d, & \text{if } x_s[n]+k\left(h_s(x_s[n])-\gamma_s(\vec{f})\right)\ge x_s^d\\ x_s[n]+k\left(h_s(x_s[n])-\gamma_s(\vec{f})\right)\end{cases} \qquad (8)$$

k in above is the step size and has a critical role in determining the convergence speed of the algorithm.

As we will see in the next section, by using adaptable step size, i.e., changing k according to the current transmission rate we can achieve high convergence rate well suited for ephemeral flows copious in high speed networks. But now we see how to relax the constraint imposed by x_s^d.

We assume that the users are greedy, i.e., they use up all available rate. Regarding this we can rewrite iterations in (8) as follows:

$$x_s[n+1] = \max\left\{0, x_s[n] + k\left(h_s(x_s[n]) - \gamma_s(\vec{f})\right)\right\} \quad (9)$$

The convergence speed can also be increased by incorporating second derivative of the cost function as [8]:

$$x_s[n+1] = \max\left\{0, x_s[n] - k\frac{h_s(x_s[n]) - \gamma_s(\vec{f})}{h_s'(x_s[n]) - \Gamma_s(\vec{f})}\right\} \quad (10)$$

where:

$$\Gamma_s(\vec{f}) \triangleq \frac{\partial^2}{\partial^2 x_s}\sum_{l=1}^{L} g_l(f_l) = \sum_{l=1}^{L}(\varphi_{sl})^2 \cdot g_l''(f_l) \quad (11)$$

In simple single path scenario we have:

$$\Gamma_s(\vec{f}) = \sum_{l\in\rho_s} g_l''(f_l) \qquad (12)$$

It is shown [8] that the common TCP congestion control can be expressed in the above framework as follows:

$$x_s[n+1] = \begin{cases} x_s[n] + a_s(x_s[n]) & \text{successful transmission} \\ \max\left\{x_s^{init}, x_s[n] - b_s(x_s[n])\right\} & \text{packet loss} \end{cases} \quad (13)$$

where:

$$a_s(x_s) = \xi\frac{\eta}{x_s}$$

and

$$b_s(x_s) = \xi \cdot x_s$$

ξ and η are protocol constants and x_s^{init} is the initial small probing rate of session s, and considered zero in some research papers. Here, the congestion measure, $\gamma_s(\vec{f})$, is merely packet loss probability of session s, which has led to the separation between successful transmission and detecting loss in (13). The corresponding reward function in this case is:

$$h_s(x_s) = \frac{\eta}{\eta + x_s^2}$$

The window adjustment algorithm would be

$$w_s[n+1] = \begin{cases} w_s + A_s(w_s[n]) & \text{successful transmission} \\ \max\left(w_s^{init}, w_s - B_s(w_s[n])\right) & \text{packet loss} \end{cases} \quad (14)$$

Where

$$A_s(w_s) = \xi\frac{\eta \cdot \tau_s^2}{w_s}$$

and

$$B(w_s) = \xi \cdot w_s$$

$$w_s^{init} = \tau_s \cdot r_s^{init}$$

Here τ_s denotes the round trip time of session s. The relation in (13) and (14) lies in the relation between window size and transmission rate, namely the Little's theorem:

$$x_s = \frac{w_s}{\tau_s}.$$

2.1 Proportional Fairness

One of the most important issues in designing a rate allocation algorithm in data networks is fairness. One notion of fairness we mainly discuss here is proportional fairness [2], [5].

The allocated rate vector $\vec{x}$ is proportionally fair if and only if it is feasible, namely $f_l \le c, \forall l$ where c is the link capacity and f_l is as defined in (1), and for any other feasible rate allocation, say $\vec{x}^*$, the average of proportional change is negative or zero:

$$\sum_{s\in\mathbf{S}}\frac{x_s^* - x_s}{x_s} \le 0 \quad (15)$$

If any session is assigned a weight ω_s and we consider rates per weight to be proportional we come up with the weighted proportional fairness:

$$\sum_{s\in\mathbf{S}}\omega_s\frac{x_s^* - x_s}{x_s} \le 0 \qquad (16)$$

In the general case we have (Ω, α)-fair rate allocation as follows [3]:

$$\sum_{s\in\mathbf{S}}\omega_s\frac{x_s^* - x_s}{x_s^\alpha} \le 0 \qquad (17)$$

where $\Omega = \{\omega_1, \omega_2, ..., \omega_S\}$. In the special case of $\omega_s = 1, \forall s$, and $\alpha = 1$, (17) reduces to proportional fairness. $\alpha = 0$ and $\alpha = 2$ correspond respectively to max-min and minimum potential delay fairness.

Kelly [2] has proposed an optimization framework for solving rate allocation problem as follows:

$$\max\sum_{s\in\mathbf{S}} U_s(x_s) \quad (18)$$

subject to:

$$\Phi \cdot \vec{x} < \mathbf{C}$$
$$x_s \geq 0$$

where $\mathbf{C} = \{c_i, i = 1, ..., L\}$ is a vector denoting link capacities and $U_s(x_s)$ is the utility function of session s sending at rate x_s.

The above formulation has much in common with (3). Kelly has shown the logarithmic utility function for users leads to a regime where rate allocations are proportionally fair. In this case the network optimization problem will be:

$$\max \sum_{s \in S} \omega_s \ln x_s \quad (19)$$

subject to

$$\Phi \cdot \vec{x} \leq \mathbf{C}$$

over

$$x_s \geq 0$$

The corresponding iteration method to the above problem is the following system of differential equations

$$\frac{d}{dt} x_s(t) = k\left(\omega_s - x_s(t) \sum_{j \in \rho_s} \mu_j(t)\right) \qquad (20)$$

where:

$$\mu_j(t) = p_j\left(\sum_{s=1}^{S} \varphi_{sj} \cdot x_s(t)\right) \qquad (21)$$

The discrete time version of this algorithm is

$$x_s[n+1] = \max\left\{0, x_s[n] + k\left(\omega_s - x_s[n] \sum_{j \in \rho_s} \mu_j[n]\right)\right\} (22)$$

where:

$$\mu_j[n] = p_j\left(\sum_{s=1}^{S} \varphi_{sj} \cdot x_s[n]\right) \qquad (23)$$

The corresponding equation to (22) in MCFC algorithm is (9). As mentioned the parameter k determines the convergence rate and using an adaptable k can increase the convergence speed as we see in the next Section. In [11], using Jacobi method, another way of increasing convergence rate is proposed.

3 High Speed Algorithm

The iterative equation (22) is shown to have desired equilibrium properties in term of fairness and stability. But the convergence rate of this algorithm should be enhanced in order to be suitable for high speed environments where short-lived flows are abundant. We consider an adaptable variable k, that updates itself in each iteration of the algorithm according to the following:

$$k_s(x_s[n]) = \frac{\upsilon}{\beta + \max_{i \in \rho_s}\left\{\left(\frac{x_s[n]}{c_i}\right)^{\theta}\right\}} = \frac{\upsilon}{\beta + \left(\frac{x_s[n]}{\min_{i \in \rho_s}\{c_i\}}\right)^{\theta}} \qquad (24)$$

The max or min term is over the entire links session s passes. In simulations we observed that this can be relaxed and considering just one link would suffice.

Therefore parameter k in (22) won't be constant any more. It will depend on iteration number n, session s and the rate this session currently sustain $x_s[n]$. We would then have:

$$x_s[n+1] = \max\left\{0, x_s[n] + k_s(x_s[n])\left(\omega_s - x_s[n] \sum_{j \in \rho_s} \mu_j[n]\right)\right\} (24)$$

Since the extra computations in this algorithm are done in end hosts, its implementation is possible while keeping the network simple and pushing the complexity to the edge, which is a desirable property.

First in a simple scenario of Fig. 3, we look at the increase in convergence rate this modification brings about. In this example two sessions share a single bottleneck link of 25 Mbps capacity. The weight parameters for these two sessions are $\omega_1 = 0.4$ and $\omega_2 = 0.3$. As we see the rates are allocated according to the weights and the desired fairness properties hold in high speed algorithm.

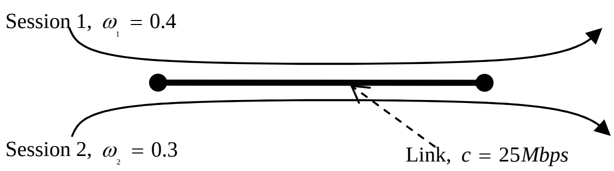

Fig. 1 simple single bottleneck scenario

Fig. 2 shows the allocated rate to sessions in conventional environment with k constant and in the proposed method. Parameters in (24) are $\beta = 0.1, \theta = 10$ and υ is varied between 0.1 to 0.4 as is shown.

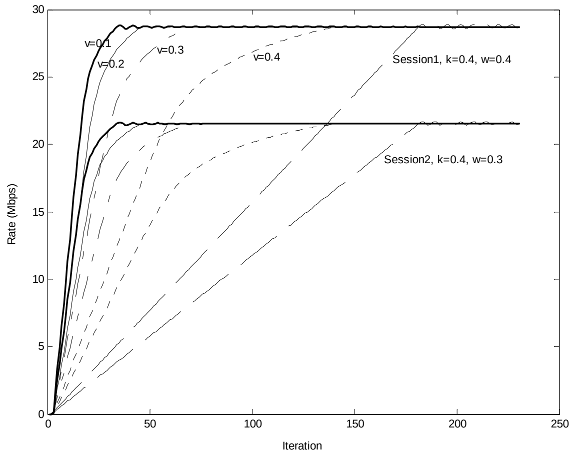

Fig. 2. Rates allocated to sessions of Fig. 3

As is clear from Fig. 2, the convergence speed is substantially increased in the adaptive method.

Now we consider example of Fig. 3, where multiple sessions interact with each other and there is multiple bottleneck links. Also in this case we consider user arrival and departure or ephemeral flows.

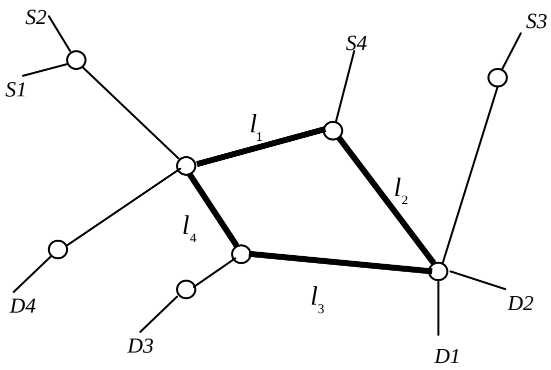

Fig. 3 Multiple session scenario sharing the same backbone

In this scenario, sessions share the same backbone and bottleneck link. The bottleneck link capacities are: c_1=50*Mbps*, c_2=30*Mbps*, c_3=60*Mbps*, c_4=70*Mbps*.The paths used by sessions is shown in Table 1.

Table1 Session paths

Session1	l_3, l_4
Session2	l_1, l_2
Session3	l_4
Session4	l_1

Fig. 4 shows the overall increase in convergence rates of all sessions. As can be seen, the equilibrium fairness is intact in high speed algorithm. Si in the figure stands for session i.

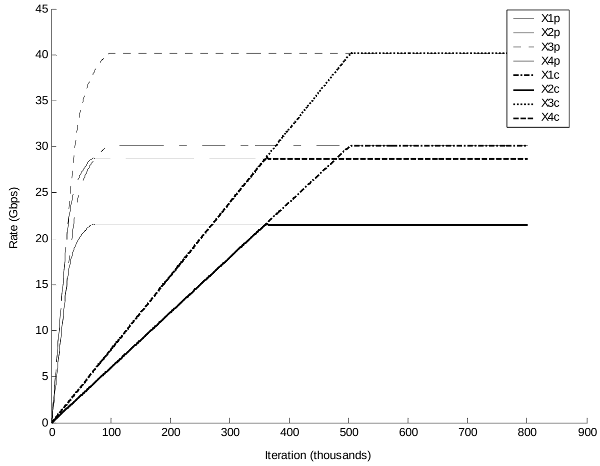

Fig. 4 Convergence rate enhancement for stable flows in scenario of Fig. 3

Now we consider the case where session 1 arrives and departs two times in the course of simulation, in other words it is an ephemeral flow. We can see that in the usual case, before the rate allocation algorithm can assign the fair rate to the session, it life is over. But we can see this problem is solved in the high speed algorithm.

Fig. 5 is the case where session 1 has short-lived traffic.

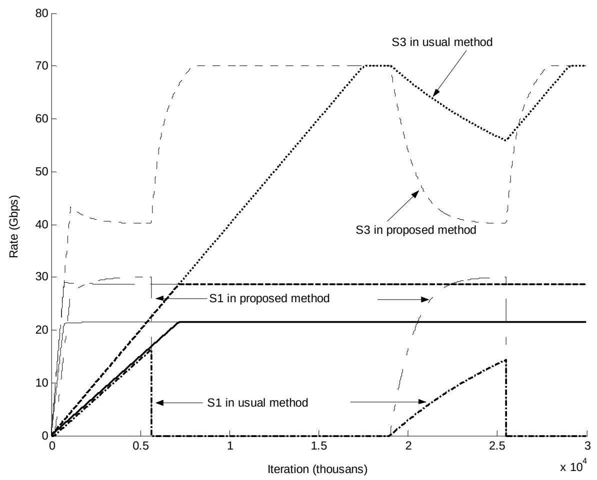

Fig. 5 Ephemeral flow of session 1

As we can see in Fig. 5, in the proposed method rates are allocated more fairly between users sharing the same bottleneck link. In this example sessions 1 and 3 share the same bottleneck link and we can see the improvement in rate allocation between these two sessions in the proposed method. After the ephemeral flow of session 1 ends, we can see the proposed method helps session 3 to faster acquire the extra bandwidth which is made available to it.

Fig. 6 to 8 shows examples of short-lived flows for sessions 2, 3 and 4 respectively. In all cases the improvement is noticeable.

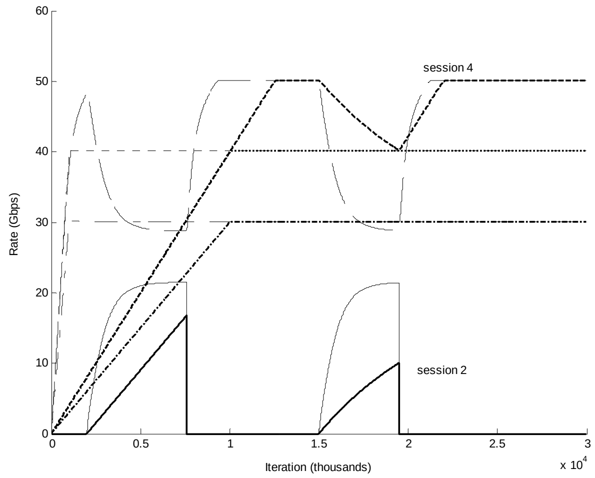

Fig. 6 Ephemeral flow of session 2

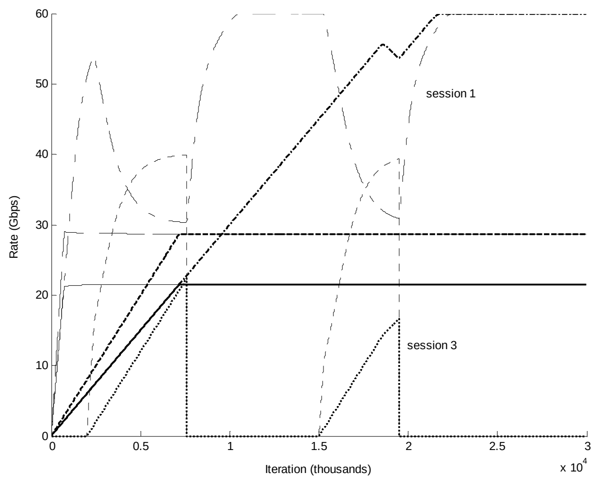

Fig. 7 Ephemeral flow of session 3

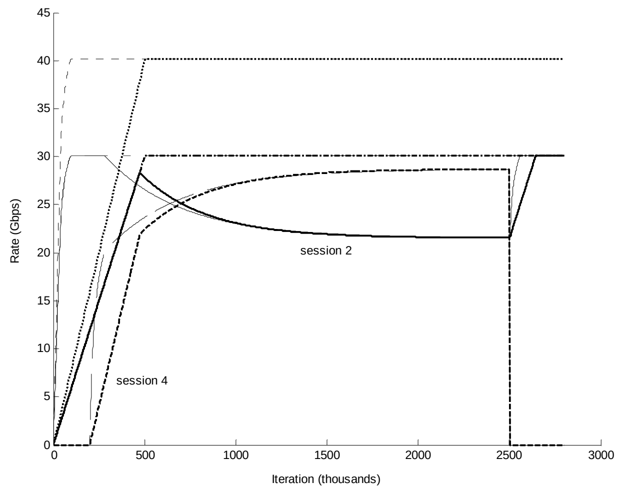

Fig. 8 Ephemeral flow of session 4

4 Conclusion

In this paper, we addressed the important problem of rate allocation algorithms, suitable for ephemeral traffic which is becoming more copious than ever as high speed networks proliferate. Furthermore we showed that this algorithm converges to the proportionally fair rates among users and the stability was demonstrated via simulation.

References:

[1] Keshav S. “An Engineering Approach to Computer Networking” Adisson-Wesley, 1977

[2] Kelly F.P., Maulloo A.K. and Tan D.K.H., “Rate control for communication networks: shadow prices, proportional fairness and stability,” J. Oper. Res. Soc., vol. 49, no. 3, pp. 237-252, Mar. 1998.

[3] Mo J. and Walrand J., “Fair End-to-End Window-Based Congestion Control,” IEEE/ACM Transactions on Networking, vol.8, pp.556-567, no. 5, Oct. 2000.

[4] Gallager, R.G. and Bertsekas, D., “Data Networks,” Prentice-Hall, NJ 1987

[5] Mazumdar R., Mason L. G., Douligeris C., “Fairness in Network Optimal Flow Control: Optimality of Product Forms,” IEEE Trans on Comm. 39, 775-782, 1991

[6] Massoulié, L. and Roberts, J., “Bandwidth sharing: objectives and algorithms,” in Proc. IEEE INFOCOM, Vol.3, 1999, pp. 1395-1403

[7] Golestani, S.J., “A Unified Theory of Flow Control and Routing in Data Communication Networks’, Ph.D Dissertation, MIT, Dec. 1979.

[8] Golestani, S.J. and Bhattacharyya, S., “End-to-End Congestion Control for the Internet: A global optimization framework.” In Proc. Of International Conf. On Network Protocols(ICNP), Oct. 1998.

[9] Lapsley, D.E. and Low, S.H., “An optimization approach to ABR control.” In Proceedings of the ICC, June 1998.

[10] Luenberger, D.G., “Linear and Nonlinear Programming,” 2nd Ed. Addison-Wesley Publishing Company, 1984.

[11] Gudarzi P., Saidi H., Sheikholeslam F., “Investigation of User Arrival and Departure in a Second-Order Proportionally-Fair Rate Allocation Algorithm,” International Workshop on Advanced Architectures and Algorithms for Internet Delivery and Applications, Orlando, Florida, USA – June 15, 2005