Proceeding of the 2004 American Control Conference
Boston, Massachusetts June 30 - July 2, 2004

A Congestion Control Algorithm for Max-Min Resource Allocation and Bounded Queue Sizes

Marios Lestas, Petros Ioannou
Electrical Engineering Department
University of Southern California
Los Angeles, California
E-mail: lestas@usc.edu

Andreas Pitsillides
Computer Science Department
University of Cyprus
Nicosia, Cyprus

***Abstract*— This paper deals with the congestion control problem in computer networks which is viewed as a resource allocation problem constrained by the additional requirement that the queue sizes need to be bounded. We propose a distributed algorithm which converges to the max-min fair allocation of resources among the users of the network and at the same time ensures that the buffers are either empty or track a reference queue size. The problem is formulated mathematically and the proposed algorithm is shown analytically to fulfil the design objectives. The local asymptotic stability of the equilibrium point is established. The problem can be viewed as a hybrid system with changing affine dynamics in different regions of the state space. The transient performance of the proposed algorithm is evaluated through simulations using Matlab. The algorithm can form the basis for the development of an end-to-end communication protocol since it requires no maintenance of per flow states within the network.** [1]

I. Introduction

The last few years, the problem of congestion control in computer networks offering a single class of best effort service, has attracted a lot of attention within the research community. The Internet is the main application which has stimulated this interest. Congestion control mechanisms currently serving the Internet are expected to lead to underutilization of the network and degradation of the quality of service provided to the users due to the increasing complexity of the system and due to increasing bandwidth-delay products. In order to resolve these problems, the research community has explored new design procedures utilizing analytical tools from the field of control theory. This design approach leads to solutions with analytically provable performance characteristics as opposed to the intuitive ad-hoc design methods where performance could only be evaluated through excessive simulations and actual implementation. Although there is still a big gap between solutions emanating from theoretical considerations and practical implementation, this gap is expected to shrink as new protocols (e.g. ECN) and emerging network technologies (e.g. UMTS Radio Access Network ,wireless and ad-hoc networks, sensor networks) offer new implementation capabilities.

The principal aim of any best-effort congestion control algorithm is to provide low-delay, low-loss services to each user and in the case where the demand for resources exceeds availability, to distribute the available resources in a fair way among the users while at the same time achieve high network utilization. Here, we consider store and forward networks such as the Internet, which offer buffering capability at each link in order to absorb statistical fluctuations of the sending rates. In order to avoid excessive delays and losses it is also necessary for the congestion control algorithm to guarantee that the queue size at each buffer is bounded. So, the congestion control problem can be decoupled in two subproblems: a queue size control problem and a resource allocation problem where the resources under consideration are the link bandwidths. In this paper, we assume deterministic models for the queueing dynamics and demonstrate how both objectives can be achieved by attempting to track a reference queue size at each link.

The paper is organized as follows. In section II we give an overview of what has been done so far from an optimization perspective, in section III we describe our model and formulate the problem mathematically, in section IV we describe the algorithm and prove all its properties, in section V we give an example and demonstrate the functionality of the algorithm through simulations and finally in section VI we offer our conclusions and future directions.

II. Fairness

As discussed in the previous section, the optimal allocation of the available resources should be characterized by high network utilization, fairness and bounded queue sizes. The problem of allocating finite resources to competitive users has been studied extensively in the fields of political science and political economics and a rich mathematical framework has been developed in order to formulate and solve the problem. Tools from this framework can be utilized to solve network problems. So, concepts like pareto-efficiency and welfare are directly related to the concepts of high network utilization and fairness respectively. In order to develop these ideas it is necessary to assign to each user a *utility* function u_i, which basically shows the preference of that user to a particular resource allocation. These individual utility functions are then aggregated in some sense through a *welfare* function $W(u_1, u_2, ..., u_n)$ which is required to be strictly increasing in all of its arguments. An allocation of the resources is then said to be fair in the sense of the welfare function chosen,

[1]This work was partly funded by IST-2001-34900 SEACORN project.

if it solves the problem of maximizing the function over the set of all feasible allocations. The optimal solution is pareto-efficient. This immediately demonstrates how the congestion control problem can be formulated as a convex optimization problem. The bounded queue size requirement does not alter the nature of the problem. It simply adds further constraints on the formulated optimization problem. A nice discussion of fairness issues is presented in [1]. In the rest of this section we analyze two fairness criteria which have dominated the literature and discuss proposed congestion control algorithms which achieve the desired fairness and at the same time aim at bounding the queue sizes.

The sum of utilities criterion corresponds to the classical utilitarian welfare function $W(u_1, u_2, ...u_n) = \sum_i u_i$. The consideration of this function in the analysis of rate control algorithms in computer networks was triggered by [2] where it was pointed out that elastic applications can be characterized by strictly increasing and concave utility functions and that the network objective is to satisfy the needs of all users by maximizing the total efficacy. It should be noted that there had been similar problem formulations long before that in [3]. Motivated by the work of Shenker the congestion control problem was then formulated as a concave maximization problem over linear constraints in [4]. The problem was then decomposed into Network and User subproblems. Primal and dual distributed algorithms were then proposed in [5] to solve relaxations of the original optimization problem which were formulated by introducing appropriate penalty functions to the original primal and dual costs. A similar approach was adopted in [6] where the original dual problem was solved using a gradient projection algorithm. The distributed nature of the solution is indeed striking. The fact that the proposed algorithm, can lead to large queue sizes and thus to excessive feedback delays led to a modification in [7] which guarantees that at equilibrium all buffers are empty. The stability of the latter was investigated in [8].

The second fairness criterion considered is the *max-min* criterion which corresponds to the Rawlsian welfare function $W(u_1, u_2, ...u_n) = min(u_1, u_2, ...u_n)$. Algorithms which achieve max-min fair allocation of sending rates among competitive users have been proposed by a number of researchers in different contexts ([3], [9], [10], [11], [12]). A very popular design procedure is to develop congestion update algorithms at each link and sending rate update algorithms at each source using the single bottleneck link case and then extend these to the multiple link case by considering at each source a feedback signal which is equal to the minimum of the congestion signals found, as a packet traverses from source to destination. Approaches which aim at bounding the queue sizes and adopt the design procedure described above, formulate the problem as a queue tracking problem for the single bottleneck link case ([13], [14], [15], [16]). In this paper we follow a different approach. It was observed in [10] that the utilitarian welfare function does not result in the same resource allocation as the Rawlsian function. In fact the difference arises when there are more than 1 bottleneck links along a particular path. Motivated by the 'absolute fairness' provided by the max-min fair allocation, we have explored ways with which the congestion control algorithms developed using the utilitarian function can be modified to produce max-min fair solutions with bounded queue sizes. It was found in [10] that if the communication among the links is changed so that the summation of prices is replaced by the maximum, max-min fair resource allocation is achieved. The problem considered in [10] can actually be viewed as a tracking problem where at each link we aim at tracking the available capacity. In this paper, we demonstrate that if instead of tracking the available capacity we attempt to track a reference queue size at each link not only do we achieve max-min allocation of resources but we also bound the queue sizes by ensuring that at each link the buffer is either empty or tracks a reference queue size. The algorithm proposed is shown analytically to have the desired characteristics and local asymptotic stability is also established.

III. Problem Formulation

We consider a packet switched, store and forward network which accommodates elastic applications. The applications are assumed to be saturated (persistent) in the sense that they always have data to send. We develop a math model for an arbitrary network in the fashion of [5]. The network consists of a finite set of sources or users $U = \{s_1, s_2, ..., s_N\}$ and a finite set of links $R = \{l_1, l_2, ..., l_L\}$, where s_i denotes user i and l_j denotes link j. Each user injects data packets into the network. The traffic is viewed as a deterministic fluid flow with continuous time dynamics and the time delays within the network are neglected. Associated with each user s_i, is its sending rate $x_i = h(q_i)$ which is chosen based on a function $h(.)$ of a feedback signal q_i that denotes the presence of congestion in the route used by user s_i. The function $h(.)$ is to be generated by the congestion control strategy. We use the vector $x = [x_1, x_2, ..., x_N]^T$ to denote all the sending rates of the sources $s_1, s_2, ..., s_N$. Similarly we use the vector $q = [q_1, q_2, ...q_N]^T$ to denote all the feedback signals of the sources. We also lump the functions $h(.)$ to form the vector valued function

$$H(q) = [h(q_1), h(q_2), ..., h(q_N)]^T \tag{1}$$

We can then write:

$$x = H(q) \tag{2}$$

To each link j we associate a buffer the queue size of which is denoted by b_j. The output capacity of the buffer is denoted by C_j. Let y_j be the flow rate of data into the buffer and let $\bar{y}_j$ be the flow rate of data out of the buffer at link j. The time evolution of the queue size is described by an ordinary differential equation of the form $\dot{b}_j = \phi(b_j, y_j, C_j)$, $b_j(0) = b_{j0}$. In this work we are assuming the following simple integrator model for the queueing dynamics:

$$\frac{db_j(t)}{dt} = Pr[y_j(t) - C_j], \qquad b_j(0) = b_{j0} \tag{3}$$

where the projection operator is defined as follows:

$$\dot{b}_j = \begin{cases} y_j - C_j & \text{if } b_j > 0 \\ y_j - C_j & \text{if } b_j = 0, y_j - C_j > 0 \\ 0 & \text{otherwise} \end{cases} \quad (4)$$

We use the vector $y = [y_1, y_2, ..., y_L]^T$ to denote all the input flow rates at links l_1 to l_N. Similarly we define the vectors $\bar{y} = [\bar{y}_1, \bar{y}_2, ..., \bar{y}_L]^T$, $b = [b_1, b_2, ..., b_L]^T$ and $C = [C_1, C_2, ...C_L]^T$. In addition, we lump the functions $\phi(.)$ to form the vector valued function

$$\Phi(b, y, C) = [\phi(b_1, y_1, C_1), ..., \phi(b_L, y_L, C_L)]^T \quad (5)$$

The queueing dynamics can then be described by the following differential equation:

$$\dot{b} = \Phi(b, y, C), b(0) = b_0 \quad (6)$$

Let $A \in R^{L \times N}$ denote the matrix that represents the route of each user. The entry in the ith row and jth column of A is denoted be a_{ij}. In this representation, A consists of elements equal to 0 or 1. If user i utilizes link j then a_{ji} is equal to 1. Otherwise it is equal to 0. If we now assume that $\bar{y} = y$ we can write the following algebraic relationship:

$$y = Ax \quad (7)$$

At each link j we associate a signal processor which generates a signal p_j which denotes the congestion status at the link. The congestion signal p_j is generated according to a control algorithm whose inputs are b_j, y_j and C_j. This control law is represented by the operator $g(.)$ such that $p_j = g(b_j, y_j, C_j)$. The operator $g(.)$ is to be determined by the congestion control scheme and it might incorporate dynamic states. We use the vector $p = [p_1, p_2, ...p_L]^T$ to denote all the congestion signals at links l_1 to l_L and we lump the operators $g(.)$ to form the vector valued operator:

$$G(b, y, C) = [g(b_1, y_1, C_1), ..., g(b_L, y_L, C_L)]^T \quad (8)$$

We can then write:

$$p = G(b, y, C), \quad (9)$$

The congestion signals generated at the links are communicated back to the sources resulting in the generation of a feedback signal q_i at each source s_i. The relationship between the feedback signals q, received at the sources and the congestion signals p, generated at the links is represented by a vector valued function $F(.)$ such that:

$$q = F(p) \quad (10)$$

The operator $F(.)$ is to be determined by the congestion control strategy. Control information can only be passed along the same routes as the data and this imposes the following mathematical constraint on the operator $F(.)$:

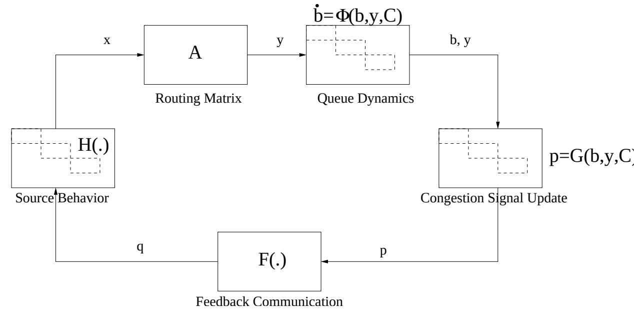


Fig. 1. Feedback System

$$F_i(.) = f(p_j | j \epsilon M_i) \ \ , M_i = \{j : a_{ji} = 1\}. \quad (11)$$

The equations indicating how the variables defined above are coupled together are summarized below:

$$\begin{aligned} Plant \quad & : y = Ax & (12) \\ & \dot{b} = \Phi(b, y, C), b(0) = b_0 & (13) \\ Controller \quad & : p = G(b, y, C), & (14) \\ & q = F(p) & (15) \\ & x = H(q) & (16) \end{aligned}$$

where $b \ \epsilon \ \Re^L$, is a state vector of the system, $x \ \epsilon \ \Re^N$, y, $p \ \epsilon \ \Re^L$, $q \ \epsilon \ \Re^N$ are system signal vectors, $\Phi : \Re^L \times \Re^L \times \Re^L \mapsto \Re^L$ is a vector field, $H : \Re^N \mapsto \Re^N$, $F : \Re^L \mapsto \Re^N$ are static, possibly nonlinear mappings and $A \epsilon \Re^{L \times N}$ is a matrix. Fig. 1 demonstrates how equations (12)-(16) are interconnected in a feedback system.

The objective is then to design the operators $H(.)$, $F(.)$ and the control law $G(.)$ such that:

$$\lim_{t \to \infty} x(t) = x^* \qquad , x^* = \max_{Ax \leq C} min(x_1, x_2, ..., x_N) \quad (17)$$

$$\lim_{t \to \infty} b(t) = b^* \qquad , b_j^* \leq b_{ref}, j = \{1, 2, ..., L\} \quad (18)$$

Equation (17) indicates that at steady state the resource allocation satisfies the max-min fairness criterion. The operators $H(.)$ and $G(.)$ are block diagonal in the sense of equations (1) and (8) and so the the desired congestion controller is said to be decentralized.

IV. Control Algorithm

We propose the following algorithm to fulfill the design objectives:

$$p = [k_p(b - b_{ref}) + w]^+, \ \frac{dw}{dt} = k_I Pr[b - b_{ref}] \quad (19)$$

$$q = A_{max}^T(p) \quad (20)$$

$$x = [K - q]^+ \quad (21)$$

where b_{ref}, k_p and k_I are design parameters, $w \epsilon \Re^L$ is a state vector, $K \epsilon \Re^N$ is a constant vector and $[z]^+ = max\{z, 0\}$. If z is a vector, the latter relationship applies for each element of the vector. The projection operator is defined in (4). If the input of the operator is a vector, (4) applies for each element of the input vector. The operator A^T_{max} is defined as follows:

$$q = A^T_{max}(p) : q_i = \max_j a_{ji} p_j \quad i = \{1, 2, ..., N\} \tag{22}$$

All the elements of the vector K are equal to k which is a design parameter. k must be greater than the maximum capacity in the network. At each link we are basically applying a simple PI controller to track the reference queue size b_{ref} and through the projection operator we make sure that all congestion signals are non-negative. The projection operator applied at the integrator within the controller ensures that the integrator state is bounded from below. This ensures that at steady state, the congestion signals at the links which do not control the sending rates are bounded. This will become apparent in the analysis. At each source we apply negative feedback and through the positive projection operator we ensure that the sending rates are non-negative. The main properties of the algorithm are outlined in the following lemmas.

Lemma 4.1: At steady state the algorithm proposed converges to a vector x^* which satisfies (17), i.e. it satisfies the max-min criterion, and to a vector b^* whose elements are equal to b_{ref} or 0.

Proof:

We are assuming that p^* is unique. It follows from (3) and (19) that at equilibrium the following are true:

$$b^*_j = b_{ref}, \;\; p^*_j > 0 \quad when \quad y^*_j = C_j \tag{23}$$

$$b^*_j = 0, \;\; p^*_j = 0 \quad when \quad y^*_j < C_j \tag{24}$$

The above can be summarized in the following equations which describe pareto optimality:

$$p^{*T}[y^* - C] = 0 \tag{25}$$
$$b^{*T}[y^* - C] = 0 \tag{26}$$

The proof for the max-min fairness of the solution is omitted due to lack of space. ∎

Lemma 4.2: The equilibrium point is locally asymptotically stable.

Proof:

Let n_{mn}, $m \neq n$, denote the number of users utilizing both links m and n. Let n_{mm} denote the number of users utilizing link m but are not utilizing link n for $n < m$. Let n_j be the total number of users utilizing link j.

At each time t, without loss of generality, change the indexing of the links so that the vector p is rearranged in descending order of its elements: $p_1 \geq p_2 \geq p_3 .... \geq p_L$. Since p_1 is the maximum value it means that all sources which utilize that link will send data with a rate of $[k - p_1]^+$. We neglect all projection operators at first and examine their effect later. So at link 1 the following equations hold:

$$y_1 = n_1(k - p_1) \tag{27}$$
$$\frac{db_1}{dt} = n_1(k - p_1) - C_1 \tag{28}$$
$$\frac{dw_1}{dt} = k_I(b_1 - b_{ref}) \tag{29}$$
$$p_1 = k_p(b_1 - b_{ref}) + w_1 \tag{30}$$

After some algebraic manipulations the above equations can be expressed in the following matrix form.

$$\frac{d}{dt}\begin{bmatrix} b_1 \\ w_1 \end{bmatrix} = \begin{bmatrix} -n_{11}k_p & -n_{11} \\ k_I & 0 \end{bmatrix}\begin{bmatrix} b_1 \\ w_1 \end{bmatrix} + \begin{bmatrix} n_1 k + n_1 k_p b_{ref} - C_1 \\ -k_I b_{ref} \end{bmatrix} \tag{31}$$

Now, all sources utilizing link 2 but not utilizing link 1 will send data with a rate of $[k - p_2]^+$. So, again neglecting the projection operators we derive the following set of equations:

$$\begin{aligned} y_2 &= n_{21}(k - p_1) + n_{22}(k - p_2) = \\ &= n_2 k - n_{21}p_1 - n_{22}p_2 \end{aligned} \tag{32}$$
$$\frac{db_2}{dt} = y_2 - C_2 \tag{33}$$
$$\frac{dw_2}{dt} = k_I(b_2 - b_{ref}) \tag{34}$$
$$p_2 = k_p(b_2 - b_{ref}) + w_2 \tag{35}$$

The above, together with the set of equations for link 1 can be put in the following matrix form:

$$\frac{d}{dt}\begin{bmatrix} b_1 \\ w_1 \\ - \\ b_2 \\ w_2 \end{bmatrix} = \begin{bmatrix} n_1 k + n_1 k_p b_{ref} - C_1 \\ -k_I b_{ref} \\ -------- \\ n_2 k + n_2 k_p b_{ref} - C_2 \\ -k_I b_{ref} \end{bmatrix} + \begin{bmatrix} -n_{11}k_p & -n_{11} & | & 0 & 0 \\ k_I & 0 & | & 0 & 0 \\ - & - & - & - & - \\ -n_{21}k_p & -n_{21} & | & -n_{22}k_p & -n_{22} \\ 0 & 0 & | & k_I & 0 \end{bmatrix}\begin{bmatrix} b_1 \\ w_1 \\ - \\ b_2 \\ w_2 \end{bmatrix} \tag{36}$$

The block lower triangular structure of the state transition matrix is evident. The procedure that we have described for the first two links, is applied up to a link m where $n_{11} + n_{22} + ... + n_{mm} = N$. Now for links r such that $r > m$, the following equations are true.

$$y_r = n_{r1}(k - p_1) + ... + n_{rm}(k - p_m) \tag{37}$$
$$\frac{db_r}{dt} = y_r - C_r \tag{38}$$
$$\frac{dw_r}{dt} = k_I(b_r - b_{ref}) \tag{39}$$
$$p_r = k_p(b_r - b_{ref}) + w_r \tag{40}$$

The above together with the set of equations derived for the links $1, 2, ..., m$ can be summarized in the following matrix equation:

$$\frac{d}{dt}\begin{bmatrix} z_1 \\ z_2 \\ \vdots \\ z_{m+1} \\ \vdots \\ z_L \end{bmatrix} = M \begin{bmatrix} z_1 \\ z_2 \\ \vdots \\ z_{m+1} \\ \vdots \\ z_L \end{bmatrix} + \begin{bmatrix} K_1 \\ K_2 \\ \vdots \\ K_{m+1} \\ \vdots \\ K_L \end{bmatrix} \tag{41}$$

where M has the form shown below:

$$\begin{bmatrix} A_{11} & 0 & 0 & 0 & 0 & \cdots \\ A_{21} & A_{22} & 0 & 0 & 0 & \cdots \\ \vdots & & & & & \\ F_{m+1,1} & \cdots & F_{m+1,m} & G_{m+1,m} & 0 & \cdots \\ \vdots & & & & & \\ F_{L,1} & \cdots & F_{L,m} & 0 & \cdots & G_{L,L} \end{bmatrix}$$

and

$$z_j = \begin{bmatrix} b_j \\ w_j \end{bmatrix}, \quad K_j = \begin{bmatrix} n_j k + n_j k_p b_{ref} - C_j \\ -k_I b_{ref} \end{bmatrix},$$

$$A_{ji} = \begin{bmatrix} -n_{ji}k_p & -n_{ji} \\ 0 & 0 \end{bmatrix} \; j \neq i,$$

$$A_{ji} = \begin{bmatrix} -n_{ji}k_p & -n_{ji} \\ k_I & 0 \end{bmatrix} \; j = i,$$

$$F_{ji} = \begin{bmatrix} -n_{ji}k_p & -n_{ji} \\ 0 & 0 \end{bmatrix}, \quad G_{ji} = \begin{bmatrix} 0 & 0 \\ k_I & 0 \end{bmatrix}$$

or in more compact form:

$$\frac{d}{dt}\begin{bmatrix} z_c \\ z_u \end{bmatrix} = \begin{bmatrix} A_c & | & 0 \\ - & - & - \\ F_u & | & G_u \end{bmatrix} \begin{bmatrix} z_c \\ z_u \end{bmatrix} + \begin{bmatrix} K_c \\ K_u \end{bmatrix} \tag{42}$$

The fact that at equilibrium, $0 < p_j^* < k$ for $j = \{1, 2, ..., m\}$ can be established by contradiction. If $p_j^* \leq 0$, then x_i^* for some i would be greater than the maximum capacity in the network which would contradict queue stability for some link j. If $p_j^* \geq k$ then $x_i^* = 0$ for some i which contradicts the max-min fair property of the solution at equilibrium. The inequality $0 < p_j^* < k$ establishes that for small perturbations of the state vector z_c about the equilibrium point, the projection operators can be neglected. So, the following is true:

$$\frac{d\delta z_c}{dt} = A_c^* \delta z_c \tag{43}$$

It was established above that A_c^* is block lower triangular and so uniform asymptotic stability is guaranteed if the eigenvalues of the diagonal matrices have negative real parts. This can be achieved by appropriate choice of the proportional and integral gains.

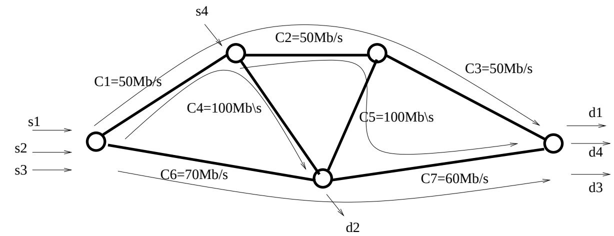


Fig. 2. Network Topology

The fact that x^* is max-min fair establishes that $y_j^* < C_j$ for $j = \{m+1, m+2, ..., L\}$. From equations (3) and (19) it can be deduced that:

$$F_u^* z_c^* + G_u^* z_u^* + K_u < 0, p_u^* = 0 \tag{44}$$

So, for any small perturbations of z the inequality is preserved. The latter together with (43) establish that the equilibrium point z^* is locally asymptotically stable. ■

Remarks: The vector z_c contains the system states $z_j \epsilon \Re^2$ for $j \epsilon \{1, 2, .., m\}$ whereas the vector z_u contains the states z_j for $j \epsilon \{m+1, ...L\}$. Similarly we define vector p_c and p_u. We state without proof that at equilibrium there exists a matrix $R^* \epsilon \Re^{N \times m}$ such that $x^* = R^* p_c^*$. This means that at equilibrium, the vector of sending rates is fully determined by p_c^*. So, the links $j \epsilon \{1, 2, ..., m\}$ are the ones controlling the sending rates at steady state. For these links p_j^* and b_j^* are strictly positive. In fact $b_j^* = b_{ref}$. For the rest of the links b_j^*, w_j^* and p_j^* are equal to 0 due to the effect of the projection operators. Another important result that follows from the local stability proof is that as long as the ordering of the vector p stays the same the matrix M is constant. This means that we can generate a state space partition $\{P_i\}_{i \epsilon I}$ and a set of constant matrices $\{D_i\}_{i \epsilon I}$ such that:

$$\dot{z}(t) = D_i z(t) + K \;\; for \;\; z(t) \epsilon P_i \tag{45}$$

This indicates that the system can be viewed as an autonomous switching system. Global asymptotic stability of such a system is currently under investigation.

V. Simulations

In this section we evaluate the performance of the proposed algorithm through simulations carried out on Matlab. The simulation scenario considers the network topology shown in Fig. 2. This topology can be represented using the A matrix and the C vector shown below. The entries in the C matrix are measured in packets/s.:

$$A^T = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \end{bmatrix},$$

$$C^T = \begin{bmatrix} 6400 & 6400 & 6400 & 12800 & 12800 & 9000 & 7700 \end{bmatrix}$$

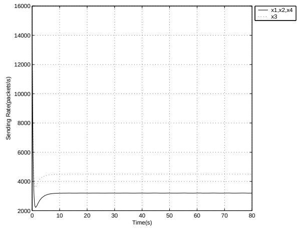
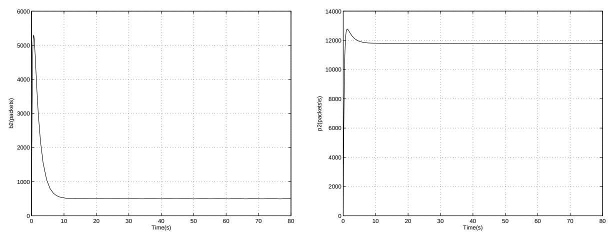


Fig. 3. Time Evolution of Sending Rates

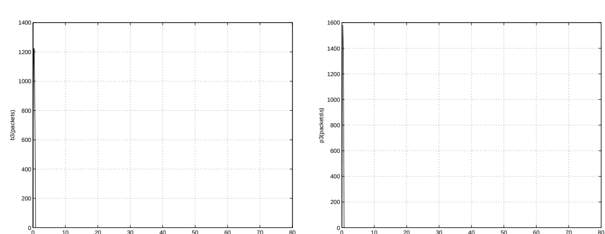


(a) Queue size (b) Congestion signal

Fig. 4. Response of the queue size and the congestion signal at link 2

The initial conditions of all integrators are set to 0 and the value of k as that appears in the algorithm proposed, is set to 15000. The proportional gain k_p and the integral gain k_i as these appear in the controller are chosen to be 2 and 1 respectively and the reference queue size is set to 500 packets. This results in closed loop poles which are real and negative and their magnitude lies between 0 and 1. It can be observed in Fig. 3 that the sending rates do converge to the max-min fair allocation of rates $x^{*T} = [3200, 3200, 4500, 3200]$.

The time evolution of the queue size b_2 and the congestion signal p_2 observed at link 2 are shown in Fig. 4, whereas the time evolution of the queue size b_3 and the congestion signal p_3 observed at link 3 are shown in Fig. 5. The former demonstrates how the links controlling the sending rates achieve good tracking of the reference queue size and result in a strictly positive congestion signal p_j, whereas the latter demonstrates how the queue sizes and the congestion signals for the rest of the links converge to zero. The system also exhibits good transient properties since good damping and speed of response are observed.



(a) Queue size (b) Congestion signal

Fig. 5. Response of the queue size and the congestion signal at link 3

VI. Conclusions and Future Work

In this paper we propose a distributed congestion control algorithm which converges to the max-min fair allocation of the available resources between competitive users in a computer network. The algorithm also guarantees that at each link the buffer is either empty or tracks a reference queue size chosen by the designer. This work was motivated by the need to find ways with which to extend the IDCC scheme which is presented in [17] to a general network topology for application in the core network of UMTS systems. The IDCC scheme considers non-linear models for the queueing dynamics at each link. So our objective is to investigate how these models and the proposed controllers can be integrated in a general network model which will lead to analytical proofs of the desired system characteristics. The no delay assumption will also be relaxed. The performance of the resulting schemes will be evaluated on the Ns simulator.

References

[1] Robert Denda, Albert Banchs, and Wolfgang Effelsberg. The fairness challenge in computer networks. *QofIS*, pages 208–220, 2000.

[2] Scott Shenker. Fundamental design issues for the future internet. *IEEE Journal on Selected Areas in Communication*, 13(7), September 1995.

[3] H. Hayden. Voice flow control in integrated packet networks. *Technical Report LIDS-TH-1152, MIT Laboratory for Information and Decision Systems*, 1981.

[4] F. Kelly. Charging and rate control for elastic traffic. *European Transactions on Telecommunications*, 8:33–37, 1997.

[5] F. Kelly, A. Maulloo, and D. Tan. Rate control in communication networks: shadow prices, proportional fairness and stability. *Journal of the Operational Research Society*, 49:237–252, 1998.

[6] S. H. Low and D. E. Lapsley. Optimization flow control — I: basic algorithm and convergence. *IEEE/ACM Transactions on Networking*, 7(6):861–874, 1999.

[7] S. Athuraliya and S. Low. Optimization flow control — II: Implementation. 1999.

[8] F. Paganini. On the stability of optimization-based flow control. In *Proceedings of 2001 American Control Conference.*, volume 6, pages 4689 –4694, 2001.

[9] Ellen L. Hahne. Round-robin scheduling for max-min fairness in data networks. *IEEE Journal of Selected Areas in Communications*, 9(7):1024–1039, 1991.

[10] B. Wydrowski and M. Zukerman. Maxnet: A congestion control architecture. *IEEE Communications Letters*, 6(11):512–514, Nov. 2002.

[11] S. Lee S. Chong and S. Kang. A simple, scalable, and stable explicit rate allocation algorithm for MAX-MIN flow control with minimum rate guarantee. *IEEE/ACM Transactions on Networking*, 9(3):322–335, 2001.

[12] S. Fahmy R. Goyal B. Vandalore S. Kalyanaraman, R. Jain. The erica switch algorithm for abr trafiic management in atm networks. *IEEE/ACM Transactions on Networking*, 8(1):87–98, 2000.

[13] G. Veciana C. F. Su and J. Walrand. Explicit rate flow control for abr services in atm networks. *IEEE/ACM Transactions on Networking*, 8(3):350–361, 2000.

[14] R. Nagarajan S. Chong and Y. Wang. Designing stable abr flow control with rate feedback and open loop control:first-order control case. *Performance Evaluation*, 34:189–206, 1998.

[15] T. Basar O. C. Imer and R. Srikant. A distributed globally convergent algorithm for fair, queue-length-based congestion control. *In Proc. IEEE Conf. Decision and Control*, 1:622–627, 2001.

[16] L. Benmohamed and S. M. Meerkov. Feedback control of congestion in packet switching networks: The case of multiple congested nodes. *International Journal of Communication Systems*, 10(5):227–246, Sept-Oct 1997.

[17] A. Pitsillides, P. Ioannou, and L. Rossides. Congestion control for differentiated-services using non-linear control theory. *In Proceedings of the Sixth IEEE Symposium on Computers and Communications*, pages 726–733, 2001.