TCP Network Calculus: The case of large delay-bandwidth product.

Eitan Altman, Konstantin Avrachenkov, Chadi Barakat

Abstract— **We present in this paper an analytical model for the calculation of network load and drop probabilities in a TCP/IP network with general topology. First we formulate our model as a nonlinear complementarity problem. Then we transform the model into two equivalent formulations: fixed point formulation and nonlinear programming formulation. These equivalent formulations provide efficient computational procedures for the solution of our model. Furthermore, with the help of the fixed point formulation we are able to prove the existence of a solution. Our model has the main advantage of not requiring the pre-definition of bottleneck links. The model also takes into account the receiver congestion window limitation. Our approach can be used for TCP/IP networks with drop tail routers as well as for TCP/IP networks with active queue management routers. We solve the problem for some network examples and we show how the distribution of load varies with network parameters. The distribution of load is sometimes counter-intuitive which cannot be detected by other models making prior assumptions on the locations of bottlenecks.**

Keywords— **TCP/IP modeling, general network topology, complementarity problem, fixed point approach, nonlinear programming.**

I. Introduction

The prediction of network behavior is an important task for a well dimensioning of network resources. A typical example of such prediction is to decide on how load will be distributed on different links of the network and how resources will be shared between the different flows. In particular, it is important to know which links will be the bottlenecks so that these links can be dimensioned properly according to the service we want to provide to users.

Most of applications in the Internet use the TCP protocol which reacts in a well known way to the loss of packets in the network [16]. In the steady state of a TCP connection, the congestion window of the protocol is increased slowly until some packets are lost and here it is divided by two to alleviate the congestion of the network which is considered as the reason behind packet losses. Given this behavior of the protocol, different models have been proposed to predict the average throughput of a TCP connection [1], [13], [20]. These models consider the network as an entity that drops packets with a certain probability. The expressions for TCP throughput together with a certain model for the network (e.g., how packets are dropped in a router for a certain rate of TCP packets) can be used to give some insights on how the network and TCP connections will behave.

In [3], the authors use a fixed-point approximation to calculate some metrics in a network crossed by long-life TCP connections and implementing Active Queue Management techniques in routers. Their model requires first the identification of bottleneck routers. An equation is written for each bottleneck router which results in a system of non-linear equations to solve. The drop probability and the average queue length in every bottleneck router as well as the throughput of the different TCP connections are calculated. In [11], the authors use the technique of stochastic differential equations to find the behavior of network traffic in the transitory regime. Again, their model requires the identification of bottleneck routers before the calculation of metrics. In [4], the authors used Markov chains as well as fixed-point approach to model one and two routers TCP/IP networks. It is not clear however if their approach can be easily extended to the case of general network topology.

Several recent papers (see [7], [8], [9], [12], [22] and references therein) have analyzed TCP-like congestion control based on the optimization of some aggregated utility function for general network topology. These models all have similarities with TCP, especially to versions based on ECN (Explicit Congestion Notification), but also differences. Discussions on the differences are given for instance in Section 4.1 in [8]. In particular, most of the above models assume that ACKs arrive continuously in time [7], [9] (or very frequently [8]). A common feature of all these models is that the utility optimization approach is related to explicit simplified dynamic evolution equations for the transmission rate of connections. Our approach, in contrast, requires as starting point only a relation between the average throughput of a connection and its average packet

All authors are with INRIA Sophia Antipolis, France, E-mails: {E.Altman,K.Avrachenkov,C.Barakat}@sophia.inria.fr.

The first author is also with C.E.S.I.M.O., Universidad de Los Andes, Merida, Venezuela.

The third author is currently with EPFL, Lausanne, Switzerland.

loss rate. The obtained results do not rely on the exact dynamics that leads to that relation, and could be applied to variants of congestion control mechanisms which need not have a linear increase and an additive decrease behavior. Another difference between our model and [7] is that we do not need to use an ECN version of TCP; in particular, our model assumes that losses occur if and only if a link is saturated. This means that the rate of acknowledgment is not a continuous function of the global throughput in the congested element, as required in [7]. In spite of the differences between our model and those based on utility optimization approach [7], [8], [9], [12], [22], it is remarkable to note that our approach also leads to a global optimization problem.

In the present paper we investigate the problem of network performance prediction without the bottleneck pre-identification requirement. First, we introduce a system of non-linear inequalities, which guarantees that the sum of TCP throughputs on each link does not exceed its capacity. We would like to note that the structure of the inequalities that we propose is simpler than those in [3], as we consider networks with large delay-bandwidth products. Then, we add *complementarity* type conditions which ensure the *automatic* localization of bottlenecks. To find a feasible point which satisfies both capacity constraint inequalities and complementarity type conditions, we use the fixed point formulation as well as the mathematical programming formulation. By using the fixed point formulation, we are able to prove the existence of a solution to our model. As a solution of our model, we obtain packet loss probabilities, the distribution of load in the network and the location of bottleneck routers. We would like to note that our model includes the possibility of having source rate limitation (e.g., the limitation imposed by TCP receiver window); this feature of TCP is not included in the above mentioned models.

Finally, we test our general approach on several benchmark network examples, for which we are able to obtain some analytical results and good approximations. In particular, the analysis of the examples shows clearly that the problem of bottleneck identification is not an easy task and that it is very sensitive to network resources distribution. For example, by slightly changing the bandwidth on a link, the bottleneck can move from a link to another link and it happens that this move is not immediate so that the two links can be bottlenecks at the same time. The change in bottleneck position alters significantly the behavior of the network. We also observed that in some cases the addition of bandwidth to some parts of the network might deteriorate the performance of other parts.

In the next section we present our TCP network model and provide methods for its solution. Then, in Section 3 we present some Benchmark network examples to show how the bottleneck position and the load distribution are sensitive to network parameters. The results of the analysis are validated via NS simulations. Finally, in the Appendix we present the second proof of the existence result.

II. TCP NETWORK MODEL

Consider a network G formed of V nodes (the routers in data network terminology). Let I be a set of groups of persistent TCP connections. We denote the source node of group $i \in I$ by S_i and its destination node by D_i, respectively. Connections of group $i \in I$ follow a fixed path $\pi_i = \{v_1^i, ..., v_{n(i)}^i\}$, where v_1^i corresponds to the first node that the connections cross after leaving the source node S_i, and $v_{n(i)}^i$ is the last node that the connections cross before reaching the destination node D_i. We also define $\pi_i(u) = \{v_1^i, ..., u\}$, that is, $\pi_i(u)$ corresponds to the part of the path π_i from the source node S_i up to node u. Of course, we are aware of the fact that the routing in the Internet is dynamic and that packets from the same TCP connection may follow different routes if some links in the network go down. We suppose that these deficiencies are not frequent and that the routing tables in Internet routers do not change during long periods of time so that our assumptions can hold. This has been shown to be the case in the Internet [21] where more than 2/3 of routes persist for days or even weeks. We also introduce the following objects:

- $M = \{\mu_1, ..., \mu_{|V|}\}$ is the capacity vector where μ_v is the capacity of node v. In reality, a capacity is associated to a link rather than a router. A router may have different output interfaces and hence different capacities. For such routers, we associate a node to each output interface. In our abstract network, we can see a node as being the part of the router where the multiple TCP connections routed via the same output interface are multiplexed together. We focus on routers where each output interface has its own buffer. The routing engine in a router decides (at a high rate) on how the different arriving packets are distributed on the different output interfaces. Packets are queued (and possibly dropped) in the buffer at the output interface before being transmitted on the link to the next router.
- $\Gamma = \{\gamma_{iv}, i \in I, v \in V\}$ is the incidence matrix, where $\gamma_{iv} = 1$ if connection i goes through node v, and is equal to zero otherwise.
- $\mathbf{p} = (p_1, ..., p_{|V|})$ is the vector of loss probabilities; p_v corresponds to the probability that a packet is lost at node v, or in other words in the buffer at the input of the link between node v and the adjacent router. We suppose here that packets from all connections are treated in the

same manner in network nodes. This can be the result of some randomization in the drop policy in router buffers (e.g., RED [14]) or the result of some randomization in the multiplexing of flows in routers (in the case of drop tail routers). Thus, we suppose that all packets are dropped with the same probability in a node and this probability is independent from that in other nodes. It follows that the probability that a packet of a connection of type i is lost in the network is equal to

$$\kappa_i = \sum_{v\in\pi_i} p_v \prod_{u\in\pi_i(v)\backslash v} (1-p_u). \quad (1)$$

- $T = (T_1, ..., T_{|I|})$ is the sending rate vector, where T_i is the sending rate of a connection of type i. The sending rate can be expressed [1], [18], [13], [20] as a function of the probability with which packets of the connection are dropped within the network.
- $N_i, i \in I$ is the number of connections of type i. Denote $[NT] = (N_1T_1, ..., N_{|I|}T_{|I|})$ the vector whose ith entry is the sending rate of all connections of type i.

We shall make the following assumptions:

A1: All links in the network have large delay-bandwidth product. This allows us to neglect queuing delays in routers, and hence, their impact on TCP throughput.

A2: The sending rate $T_i(\kappa_i)$ is a continuous function of the packet loss probability κ_i.

We shall consider in particular some well known forms of relations between loss probabilities and throughput. The following expression (so-called "square root formula" [20]) is well suited for a small number of timeout events, which is typical in large delay-bandwidth product networks

$$T_i(\kappa_i) = MSS_i \min\{\frac{1}{\theta_i}\sqrt{\frac{c_i}{\kappa_i}}, \frac{W^i_{max}}{\theta_i}\}, \quad i \in I, \quad (2)$$

where MSS_i is the maximal segment size, W^i_{max} is the receiver window size, θ_i is the average round-trip time of the connection and c_i is a constant that depends on the version of the TCP implementation and on the characteristics of the process of inter-loss times [1]. For example, if we assume that inter-loss times are exponentially distributed and the delay ACK mechanism is disabled, then $c_i = 2$ [1].

The next expression [13] (which we shall refer to as "PFTK formula") is known to be more suitable when the timeout probabilities are not negligible:

$$T_i(\kappa_i) = \begin{cases} MSS_i \dfrac{\frac{1-\kappa_i}{\kappa_i} + W(\kappa_i) + Q(\kappa_i, W(\kappa_i))}{\theta_i(\frac{b_i}{2}W(\kappa_i)+1) + \frac{Q(\kappa_i,W(\kappa_i))F(\kappa_i)T^i_0}{1-\kappa_i}}, \\ \text{if } W(\kappa_i) < W^i_{max}, \\ MSS_i \dfrac{\frac{1-\kappa_i}{\kappa_i} + W^i_{max} + Q(\kappa_i, W^i_{max})}{\theta_i(\frac{b_i}{8}W^i_{max} + \frac{1-\kappa_i}{\kappa_i W^i_{max}} + 2) + \frac{Q(\kappa_i,W^i_{max})F(\kappa_i)T^i_0}{1-\kappa_i}}, \\ \text{otherwise,} \end{cases} \quad (3)$$

where

$$\begin{aligned} W(q) &= 2/3 + 2\sqrt{(1-q)/(3q) + 1/9}, \\ Q(q,w) &= \min\{1, (1-(1-q)^3)(1+(1-q)^3 \times \\ &\quad \times (1-(1-q)^{w-3}))/(1-(1-q)^w)\}, \\ F(q) &= 1 + q + 2q^2 + 4q^3 + 8q^4 + 16q^5 + 32q^6, \end{aligned}$$

and where b_i is the number of packets acknowledged by an ACK, and T^i_0 is the basic timeout duration.

It is clear that the capacity at each node cannot be exceeded by the rate of packets that cross it. This leads to the following system of inequalities

$$\sum_{i\in I} \gamma_{iv} \left(\prod_{u\in\pi_i(v)} (1-p_u) \right) N_i T_i(\kappa_i) \le \mu_v, \quad v \in V. \quad (4)$$

where the left-hand term represents the sending rate of TCP connections crossing node v reduced by the number of packets dropped before reaching the output interface of v.

Thus, we have obtained a system of $|V|$ inequalities for $|V|$ unknowns $p_1, ..., p_{|V|}$ that we have to solve in order to model the performance of TCP connections and the distribution of load on network nodes. First, let us show that this system of inequalities is feasible.

Proposition 1: Under A2, the system of inequalities (4) is feasible. Moreover, there is a continuum of feasible solutions.

Proof: There is an obvious feasible solution: $p_v = 1, \forall v \in V$, which results in a strict inequality in (4). Since this point is interior, and κ_i are continuous in the p_v's and consequently T_i's are continuous in the κ_i's, there is a feasible region with nonzero measure. ■

Even though there is a continuum of feasible solutions to (4), most of them do not correspond to a real TCP network state. An example of such solutions is a one that gives high drop probabilities so that all nodes are poorly utilized. On contrary, TCP is designed to fully utilize the available resources of the network. We observed from numerous TCP network simulations carried out with the help of the network simulator NS [17] that a link can be either bottleneck with a substantial amount of packet losses at its input, or it can be underutilized with negligible packet-loss probability. These two states of a link are quite mutually exclusive. The latter observation leads us to the following *complementarity* type conditions

$$p_v \left(\mu_v - \sum_{i \in I} \gamma_{iv} \left(\prod_{u \in \pi_i(v)} (1 - p_u) \right) N_i T_i(\kappa_i) \right) = 0, \tag{5}$$

for $v \in V$. These conditions say that packets are only dropped in nodes which are fully utilized. These are the bottleneck nodes that limit the performance of TCP connections. Other nodes are well dimensioned so that they do not drop packets and thus they do not impact the performance of TCP.

We shall refer to the system of (4) and (5), plus the natural condition

$$0 \le p_v \le 1, \quad v \in V, \tag{6}$$

as the *Complementarity Problem Formulation* (CP formulation). Next we show that the Complementarity Problem Formulation is equivalent to a *Fixed Point Formulation* (FP formulation).

Lemma 1: The CP formulation (4), (5) and (6) is equivalent to the following Fixed Point formulation

$$p_v = \text{Pr}_{[0,1]} \Big\{ p_v - \alpha \Big(\mu_v - \sum_{i \in I} \gamma_{iv} \Big(\prod_{u \in \pi_i(v)} (1 - p_u) \Big) N_i T_i(\mathbf{p}) \Big) \Big\}, \tag{7}$$

where $\alpha > 0$ and $\text{Pr}_{[0,1]}\{x\}$ is the projection on the interval $[0, 1]$, that is,

$$\text{Pr}_{[0,1]}\{x\} = \begin{cases} 0, & x < 0, \\ x, & 0 \le x \le 1, \\ 1, & 1 < x. \end{cases}$$

Proof: First let us prove that any solution of CP is a solution of FP. Take any $v \in V$. According to the complementarity condition (5), if the inequality (4) is strict, $p_v = 0$. Hence, we have

$$0 = \text{Pr}_{[0,1]}\{-\alpha(\mu_v - \sum_{i \in I} \gamma_{iv}(\prod_{u \in \pi_i(v)} (1 - p_u))N_i T_i(\mathbf{p}))\},$$

and consequently p_v satisfies (7). Now, if $p_v > 0$,

$$\Delta_v := \mu_v - \sum_{i \in I} \gamma_{iv}(\prod_{u \in \pi_i(v)} (1 - p_u))N_i T_i(\mathbf{p})) = 0$$

we have $p_v = \text{Pr}_{[0,1]}\{p_v\}$, that is true, since $p_v \in [0, 1]$. In case both $p_v = 0$ and $\Delta_v = 0$, the equality (7) holds trivially.

Next let us show that any solution of FP is also a solution of CP. The condition (6) follows immediately from the definition of the projection. Next we show that the inequality (4) holds. Suppose on contrary that $\Delta_v < 0$. Then, it follows from (7) that p_v is necessarily equal to one. However, if $p_v = 1$, $\Delta_v = \mu_v > 0$. Thus, we came to the contradiction and hence (4) holds. Finally, we need to show that the complementarity condition (5) holds, namely, we need to show that it is not possible to have $p_v > 0$ and $\Delta_v > 0$ simultaneously. Suppose on contrary that these two strict inequalities hold. The inequality $\Delta_v > 0$ implies that $p_v - \alpha\Delta_v < 1$. Hence, according to (7),

$$p_v = p_v - \alpha\Delta_v.$$

The latter implies that $\Delta_v = 0$, which is the contradiction. Thus, the complementarity condition (5) holds as well. This completes the proof. ■

Now, using the FP formulation, we are able to prove the existence of a solution to our model.

Theorem 1: The TCP network model (4), (5) and (6) has a solution.

Proof: From Lemma 1 we know that the TCP network model (4), (5) and (6) is equivalent to the Fixed Point formulation (7). Under Assumption A2, the mapping (7) is well-defined and continuous on the compact and convex set $\times_{v \in V}[0, 1]$. Furthermore, (7) maps the set $\times_{v \in V}[0, 1]$ into itself. Hence, all conditions of Brouwer Fixed Point Theorem [6], [19] are satisfied and we can conclude that the system of (4), (5) and (6) has a solution. ■

Note that the FP formulation provides not only the theoretical means to prove the existence of a solution to our model, but it also gives a practical algorithm for its calculation. Namely, we can calculate a solution by using the following fixed point iterations

$$p_v^{(k+1)} = \text{Pr}_{[0,1]} \Big\{ p_v^{(k)} - \alpha \left(\mu_v - \sum_{i \in I} \gamma_{iv} \left(\prod_{u \in \pi_i(v)} (1 - p_u^{(k)}) \right) N_i T_i(\mathbf{p}^{(k)}) \right) \Big\}, \tag{8}$$

where α is a parameter that can be chosen to control stability and the speed of convergence.

Next we propose yet another formulation which also leads to an efficient computational algorithm for the solution of the system (4), (5) and (6). This third formulation is based on the application of the nonlinear mathematical programming to complementarity problems [5]. Therefore, we shall refer to it as *Nonlinear Programming formulation* (NP formulation). Let us consider the following nonlinear mathematical program

$$\min \sum_{v \in V} p_v z_v \tag{9}$$

subject to

$$\sum_{i \in I} \gamma_{iv} \left(\prod_{u \in \pi_i(v)} (1 - p_u) \right) N_i T_i(\mathbf{p}) + z_v = \mu_v,$$

$$0 \le z_v, \quad 0 \le p_v \le 1, \quad v \in V.$$

Note that variables z_v play the same role as the supplementary variables introduced in linear programming. They transform a system of inequalities into a system of equations. The intuition behind the mathematical program (9) can be explained as follows: we start from a feasible point inside the region defined by inequalities (4), and then, by minimizing $\sum_{v \in V} p_v z_v$, we try to satisfy the complementarity conditions (5). Since in (9) we minimize a continuous function over a compact set, this program has a global minimum. Furthermore, the value of the objective function evaluated at this minimum is zero if and only if the original system (4), (5), (6) has a solution. Thus, due to Theorem 1, the mathematical program (9) provides a solution to the system (4), (5), (6).

We would like to emphasize that the main advantage of using either FP formulation or NP formulation is that one does not need to care as in [3] about locating bottleneck nodes in order to establish a system of equations that solves the problem. If there is no *a priori* information on the location of the bottlenecks, then one can need to check up to $2^{|V|}$ cases. As we shall see later in the section on the Benchmark examples, the localization of bottleneck nodes is not always so intuitive. A small change in network parameters may shift the bottleneck from one node to another.

Note that for TCP/IP networks with high delay-bandwidth products, the packet loss probabilities p_v are typically small (connections operate at large windows). Therefore, we can simplify our model even further. Equations (1) and (4) take now the form

$$\kappa_i = \sum_{v \in \pi_i} p_v,$$

$$\sum_{i \in I} \gamma_{iv} N_i T_i \le \mu_i.$$

As an example, if we use the square root formula for TCP throughput, we obtain the following system of equations and inequalities

$$\sum_{i \in I} \gamma_{iv} \frac{k_i}{\sqrt{\sum_{v \in \pi_i} p_v}} \le \mu_i, \quad v \in V, \tag{10}$$

$$p_v \left(\mu_v - \sum_{i \in I} \gamma_{iv} \frac{k_i}{\sqrt{\sum_{v \in \pi_i} p_v}} \right) = 0, \quad v \in V, \tag{11}$$

where we denote $N_i \sqrt{c}/\theta_i$ by k_i to simplify notations. In the sequel, we shall refer to the above system as the *rough approximation model*. Note that the rough approximation model can be written in an elegant form by using the matrix notations introduced in the beginning of the present section. Namely, the inequalities and the equations for the rough approximation model can be written as follows:

$$[NT]\Gamma \le M, \quad [M - [NT]\Gamma]_v p_v = \mu_v.$$

Remark 1: There are arguments in favor of infinite W^i_{max}. In this case, we allow TCP to load the network as much as possible without any limitation from the side of the receiver. Clearly it is important to model such a situation as well. Note that if we take $W^i_{max} = \infty$, the Assumption A2 will be violated at points $\kappa_i = 0$. However, if one chooses $W^i_{max} = \max_{v \in V}\{\mu_v\}$, the Assumption A2 holds and TCP rates are only bounded by the network resources.

III. Benchmark examples

In this section we present several benchmark examples. Even though we have succeeded to prove the existence of a solution to our model, the uniqueness is still an open problem. We are able to show the uniqueness for some benchmark examples. We compare the analytical results and approximations with the fixed point iterations (8), the numerical solution of mathematical program (9) and with simulations obtained via NS. Actually, the numerical solutions obtained via (8) and (9) coincide within the computer precision. However, we would like to note that the method of fixed point iterations achieves the solution much faster in comparison with (9). We have chosen the parameters of the simulations so that to avoid timeouts. Thus, we could use the simple square root formula (2) for TCP throughput. We are interested in the case when TCP has no restrictions on its throughput other than the network capacity. Hence, we take $W^i_{max} = \infty$ for our analytical calculations and $W^i_{max} = \max_{v \in V}\{\mu_v\}$ for our numerical calculations. In all our experiments, we have used the New Reno TCP version and we have set $MSS_i = 512 Bytes$. For routers, we have chosen RED as queue management policy with the following parameters: min_thresh=10 packets, max_thresh=30 packets, queue_limit=50, p_max=0.1, and averaging_weight=0.002.

A. One node case

For completeness of the presentation let us consider a single node example. Namely, let m different type TCP connections cross a single node. In the case of the rough approximation model, we have the following equation for the packet loss probability

$$\sum_{i=1}^{m} \frac{N_i}{\theta_i} \sqrt{\frac{c_i}{p}} = \mu.$$

Clearly, the above equation always has a unique solution which is given by

$$p_* = \frac{1}{\mu^2} \left(\sum_{i=1}^{m} \frac{N_i \sqrt{c_i}}{\theta_i} \right)^2 . \quad (12)$$

Note that if the delay-bandwidth products are large ($\mu\theta_i >> 1$), the above formula gives a small packet loss probability. We may expect that the rough approximation model and the following more precise model have close solutions

$$\sum_{i=1}^{m} \frac{N_i}{\theta_i} \sqrt{\frac{c_i}{p}} (1-p) = \mu.$$

The above equation leads to the following equivalent quadratic equation

$$p^2 - (\frac{1}{p_*} + 2)p + 1 = 0,$$

with p_* as in (12). It has two roots:

$$p_{1,2} = \frac{1}{2} \left((\frac{1}{p_*} + 2) \pm \sqrt{\frac{1}{p_*}(\frac{1}{p_*} + 4)} \right).$$

The root corresponding to "+" in the above expression is greater than one. Therefore we choose the root corresponding to "−". For small values of p_* this root has the following asymptotics

$$p = p_* - 2p_*^2 + o(p_*^2).$$

Thus, we can see that indeed for the case of large delay-bandwidth products the rough approximation model gives results that are very close to the ones of the original model (4), (5) and (6). In particular, the two models have unique solutions.

B. Simple two node tandem network

Let a group of N TCP connections of the same type successively cross two nodes with capacities μ_1 and μ_2 (see Figure 1). We denote the probability of packet loss at the first node by p_1 and the probability of packet loss at the second node by p_2. From (2) the sending rate of a TCP connection is given by

$$T(p_1, p_2) = \frac{1}{\theta} \sqrt{\frac{c}{p_1 + (1-p_1)p_2}}.$$

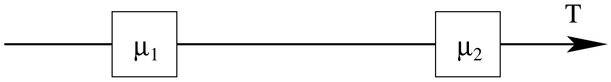

Fig. 1. Topology for Example 2

Then, according to (4), we have

$$T(p_1,p_2)(1-p_1) \le \mu_1, \;\; T(p_1,p_2)(1-p_1)(1-p_2) \le \mu_2. \quad (13)$$

where $k = N\sqrt{c}/\theta$. The complementarity conditions (5) take the form

$$p_1 \left(\mu_1 - T(p_1,p_2)(1-p_1)\right) = 0,$$
$$p_2 \left(\mu_2 - T(p_1,p_2)(1-p_1)(1-p_2)\right) = 0. \quad (14)$$

First let us consider the rough approximation model:

$$\frac{k}{\sqrt{p_1+p_2}} \le \mu_1, \quad \frac{k}{\sqrt{p_1+p_2}} \le \mu_2, \quad (15)$$

$$p_1 \left(\mu_1 - \frac{k}{\sqrt{p_1+p_2}} \right) = 0, \; p_2 \left(\mu_2 - \frac{k}{\sqrt{p_1+p_2}} \right) = 0. \quad (16)$$

We note that for the rough approximation model, the analysis of the two cases: $\mu_1 < \mu_2$ and $\mu_1 > \mu_2$ is the same. Let us consider for example the case $\mu_1 < \mu_2$. Clearly,

$$\frac{k}{\sqrt{p_1+p_2}} < \mu_2$$

and hence from complementarity conditions (16), we conclude that $p_2 = 0$. The first inequality in (15) becomes equality. The latter leads to the expression for the packet loss probability in the first node.

$$p_1 = \frac{k^2}{\mu_1^2} = \frac{N^2 c}{\mu_1^2 \theta^2}$$

Now let us consider the case $\mu_1 = \mu_2 = \mu$. Inequalities (15), which become equalities, and conditions (16) are now satisfied for all p_1 and p_2 such that $p_1 + p_2 = N^2c/\mu^2\theta^2$. That is, the rough approximation model has a non unique solution if $\mu_1 = \mu_2$.

Next we analyze the more precise model (13),(14). In particular, we shall see that (13) and (14) always possess a unique solution. First we consider the case $\mu_1 \le \mu_2$. According to conditions (14), there could be three possibilities: (a) only the second node is a bottleneck ($p_1 = 0, p_2 > 0$); (b) both nodes are bottlenecks ($p_1 > 0, p_2 > 0$); and (c) only the first node is a bottleneck ($p_1 > 0, p_2 = 0$). In case (a), (13) and (14) imply

$$\frac{k}{\sqrt{p_2}} \le \mu_1, \quad \frac{k}{\sqrt{p_2}}(1-p_2) = \mu_2.$$

The above inequality and equation lead to

$$\mu_2 = \frac{k}{\sqrt{p_2}}(1-p_2) \le \mu_1(1-p_2) < \mu_1.$$

The latter means that $\mu_2 < \mu_1$, which is the contradiction, and hence possibility (a) cannot be realized. In case (b), according to complementarity conditions (14), inequalities (13) become equalities which lead to

$$\mu_2 = \mu_1(1-p_2) < \mu_1.$$

This is again the contradiction, and consequently, possibility (b) cannot be realized as well. Only possibility (c) is left. In this case, (13) and (14) imply

$$\frac{k}{\sqrt{p_1}}(1-p_1) = \mu_1, \tag{17}$$

$$\frac{k}{\sqrt{p_1}}(1-p_1) \le \mu_2. \tag{18}$$

If equation (17) has a solution, inequality (18) is satisfied as $\mu_1 \le \mu_2$. The existence and uniqueness of a solution to (17) has been shown in the previous subsection. Therefore, the system (13),(14) has a unique solution if $\mu_1 \le \mu_2$. In particular, we conclude that in this case the first node is a bottleneck.

The case $\mu_1 > \mu_2$ is more difficult to analyze. It turns out that if we set $\mu_1 = \mu_2 = \mu$ and we start to increase the value of μ_1, then initially there will be an interval (μ, μ^*) inside which there is a solution to the system of equations

$$T(p_1,p_2)(1-p_1) = \mu_1, \;\; T(p_1,p_2)(1-p_1)(1-p_2) = \mu_2, \tag{19}$$

with both p_1 and p_2 positive, and then for the interval $[\mu^*, \infty)$, the second node becomes a bottleneck ($p_1 = 0$). To analyze this phenomenon, one can directly solve the system of equations (19) for the interval (μ, μ^*). However, it is simpler to use the "perturbation approach". Take $\mu_1 = \mu + \varepsilon$ and $\mu_2 = \mu$ and look for the solution of the system (19) in the following form $p_1(\varepsilon) = p_1^* + q_1\varepsilon + ...$ and $p_2(\varepsilon) = q_2\varepsilon + ...$. p_1^* is the solution of equation (17) and q_1 and q_2 are two coefficients to calculate. After the substitution of these series into equations (19), expanding nonlinear expressions as power series and collecting terms with the same power of ε, we obtain the next system for the first order approximation

$$q_1 + (1-p_1^*)q_2 = 0,$$

$$(1+p_1^*)q_1 + (1-p_1^*)^2 q_2 = -\frac{2p_1^*\sqrt{p_1^*}}{k}.$$

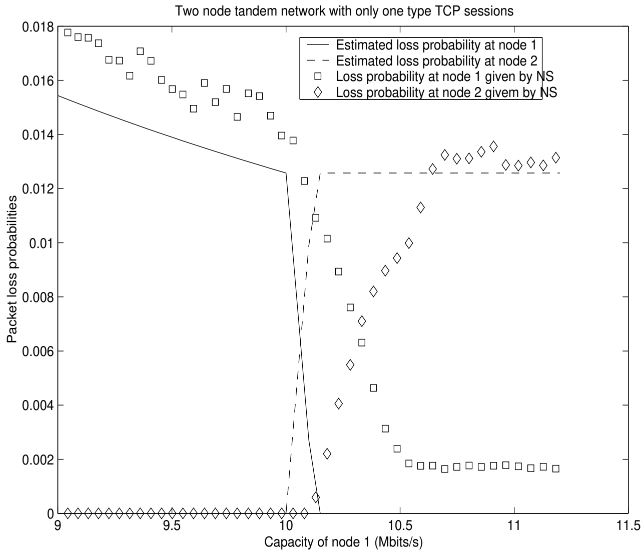

Fig. 2. Simple two node tandem network

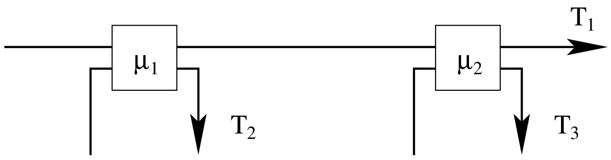

Fig. 3. Topology for Example 3

The solution of the above equations gives

$$p_1(\varepsilon) = p_1^* - \frac{\sqrt{p_1^*}}{k}\varepsilon + ..., \tag{20}$$

$$p_2(\varepsilon) = \frac{\sqrt{p_1^*}}{(1-p_1^*)k}\varepsilon + ... = \frac{\varepsilon}{\mu} + ... \tag{21}$$

Using the approximate expression for $p_1(\varepsilon)$, we can estimate μ^*. Namely, $\mu^* = \mu + \varepsilon^*$, where $\varepsilon^* = k\sqrt{p_1^*}$.

By using either the fixed point iteration method (8) or the nonlinear programming (9) we can obtain the packet loss probabilities p_1 and p_2 (see Figure 2). At the same Figure 2 we also plot the packet loss probabilities obtained by NS. The following parameters were used: $N = 40$, $\theta = 204ms$, $\mu = 10Mbits/s$.

We would like to note that the analytical approximations (20) and (21) are so good that they cannot be distinguished from the plots obtained via (8) or (9).

C. Two node network with cross traffic

Next we consider a two node tandem network with cross traffic (see Figure 3).

Let us show that both nodes in this example are bottlenecks. Namely, we need to show that the following system of equations always has a solution

$$\frac{k_1}{\sqrt{p_1+p_2}}+\frac{k_2}{\sqrt{p_1}}=\mu_1, \tag{22}$$

$$\frac{k_1}{\sqrt{p_1+p_2}}+\frac{k_3}{\sqrt{p_2}}=\mu_2, \tag{23}$$

where $k_i = N_i\sqrt{c_i}/\theta_i$. Here we first analyze the rough approximation model. Later on we shall show that the refined approximation model gives practically the same results as the rough approximation model. The system (22),(23) is equivalent to the following set of equations

$$\frac{k_1}{\sqrt{p_1+p_2}}=x, \quad \frac{k_2}{\sqrt{p_1}}=\mu_1-x, \quad \frac{k_3}{\sqrt{p_2}}=\mu_2-x.$$

In turn, the above set of equations gives the following single equation for unknown x.

$$\frac{k_1}{\sqrt{\dfrac{k_2^2}{(\mu_1-x)^2}+\dfrac{k_3^2}{(\mu_2-x)^2}}}=x \tag{24}$$

Denote the left hand side of this equation by $f(x)$. Next, we prove that the graph of $y = f(x)$ intersects the line $y = x$ only at one point (see Figure 4). Towards this end, we compute the derivative

$$f'(x)=-\frac{k_1(\dfrac{k_2^2}{(\mu_1-x)^3}+\dfrac{k_3^2}{(\mu_2-x)^3})}{(\dfrac{k_2^2}{(\mu_1-x)^2}+\dfrac{k_3^2}{(\mu_2-x)^2})^{3/2}}$$

and observe that it is negative for $x \in [0, \min\{\mu_1, \mu_2\})$. Hence, the function $f(x)$ is monotonous on the interval $[0, \min\{\mu_1, \mu_2\})$, and consequently, equation (24) always has a unique solution. The latter implies that the system (22),(23) has a unique solution as well. Note that the system (22),(23) can be solved via the direct application of Newton type methods for the solution of nonlinear systems.

Let us now consider a particular symmetric case when we are able to obtain exact analytical expressions. Let $\mu_1 = \mu_2 = \mu$, $\theta_1 = \theta_2 = \theta_3 =: \theta$ and $N_1 = N_2 = N_3 =: N$. Clearly, in this case $p_1 = p_2 = p$. After straightforward calculations, we get

$$p=\frac{(1+\sqrt{2})^2}{2}\frac{cN^2}{(\theta\mu)^2}=\frac{3+2\sqrt{2}}{2}\frac{cN^2}{(\theta\mu)^2}.$$

We also obtain

$$\frac{T_2}{T_1}=\frac{T_3}{T_1}=\sqrt{2}.$$

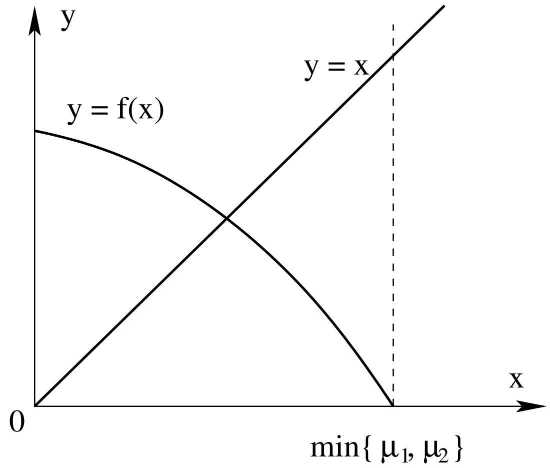

Fig. 4. Uniqueness of the solution for Example 3

This symmetric two node network with cross traffic was analyzed in [15] to study TCP fairness. In particular, in [15], the ratio T_2/T_1 is estimated as 1.5. Thus, we can see that our model agrees well with previous observations. The fact that $T_2/T_1 \approx 1.5$ means that TCP fairness is between max-min fairness ($T_2/T_1 = 1$) and proportional fairness ($T_2/T_1 = 2$) [2], [7], [10].

Next we study a non symmetric case. We fix $\mu_2 = \mu = 10Mbits/s$ and we vary the value of μ_1 (The other parameters are $N_1 = N_2 = N_3 = 20$, $\theta_1 = \theta_2 = \theta_3 = 204ms$). We plot the packet loss probabilities p_1, p_2 and the values of throughputs T_1, T_2, T_3 with respect to the ratio μ_1/μ_2 in Figures 5 and Figure 6, respectively. Note that if we increase μ_1 from the value μ and keep μ_2 unchanged, the throughput of connection 3 is deteriorated. At the first sight, this fact might appear to be surprising, as we are only increasing the total capacity of the network. However, we can propose the following explanation for this phenomenon: with the increase of the capacity of node 1, the throughput of type 1 connections increases as well; the latter creates an additional load on node 2, which leads to the deterioration in the performance of connections of type 3.

Finally, we have plotted the same graphs for the more precise model (4),(5) and it turns out that for the set of the network parameters under consideration, the results from the rough approximation model and the results from the more precise model (4),(5) are practically indistinguishable. The figures also show graphs from NS simulations which validate our modeling results.

D. Three node tandem network

Finally let us consider a three node tandem network (see Figure 7). We set the following values of the parameters:

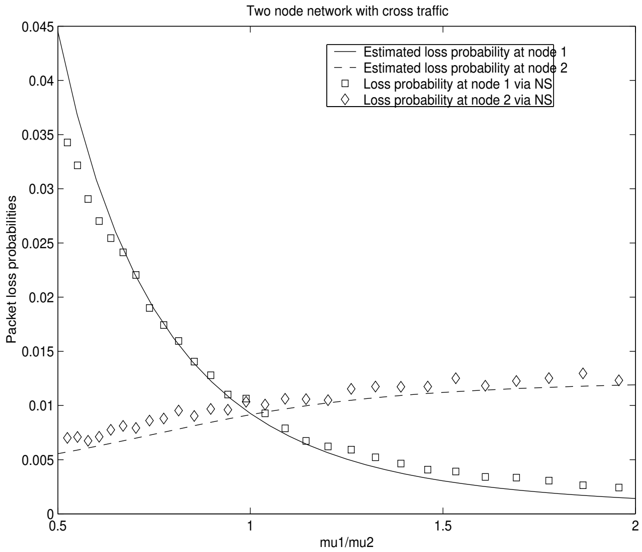

Fig. 5. Two node network with cross traffic

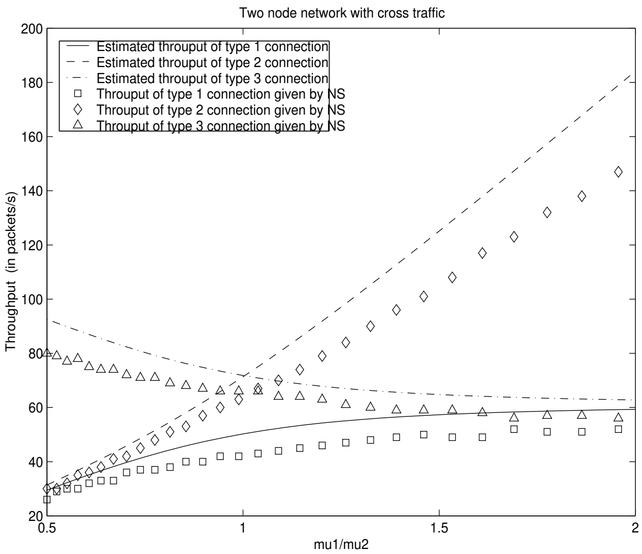

Fig. 6. Two node network with cross traffic

$\theta_1 = 304ms$, $\theta_2 = \theta_3 = 204ms$, $N_1 = N_2 = N_3 = 20$, $\mu_1 = 12Mbits/s$, $\mu_3 = 8Mbits/s$ and we vary capacity μ_2 over the range [10Mbits/s;22Mbits/s]. In Figures 8 and 9, we plot packet loss probabilities p_1, p_2, p_3, and sending rates T_1, T_2, T_3, respectively. The probabilities are calculated with the help of the fixed point iteration method (8). The plots show that first only the node 2 is a bottleneck (we call it, phase 1), then node 3 also becomes a bottleneck (phase 2), then with the further increase in the value of μ_2, all three nodes become bottlenecks (phase 3), and finally for large values of μ_2 only nodes 1 and 3 are left as bottlenecks (phase 4). Even though this sequence of changes in the network is quite intuitive, it is practically impossible to relay on intuition to predict the boundaries

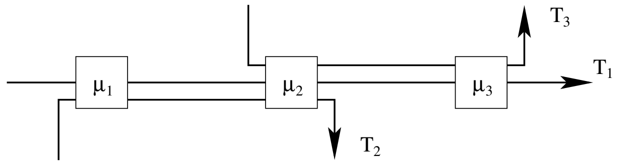

Fig. 7. Topology for Example 4

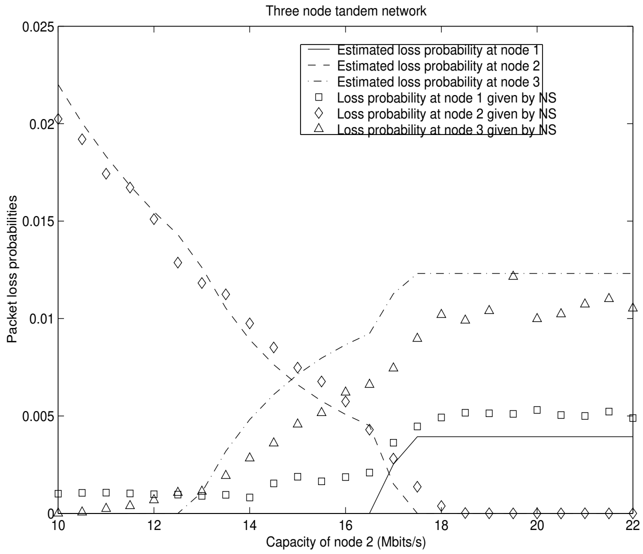

Fig. 8. Three node tandem network

for these phases. This fact highlights utility of the formal approaches such as FP and NP formulations.

The non monotonous behavior of the sending rate T_1 is another interesting fact. We have already noticed such behavior in the previous Example 3; the increase of the capacity in one part of the network can sometimes lead to the decrease of throughputs of some TCP connections. We also note that the previous examples of one node network and two node tandem network with cross traffic, are the limiting cases of this more general topology and can be used to construct approximations when either $\mu_2 << \mu_1, \mu_3$ or $\mu_2 >> \mu_1, \mu_3$.

IV. CONCLUSIONS

Several approaches exist for modeling and analyzing TCP/IP networks. On one hand, there is the approach that focuses on a single TCP connection and calculates its throughput as a function of the characteristics of the packet loss process. On the other hand, there are approaches that consider the whole network and try to predict the throughput of all connections simultaneously, taking into account their mutual interaction. This paper belongs to the second research direction. We proposed a model for the network and we presented three equivalent formulations (CP, FP and NP) of it. In particular, FP and NP formulations

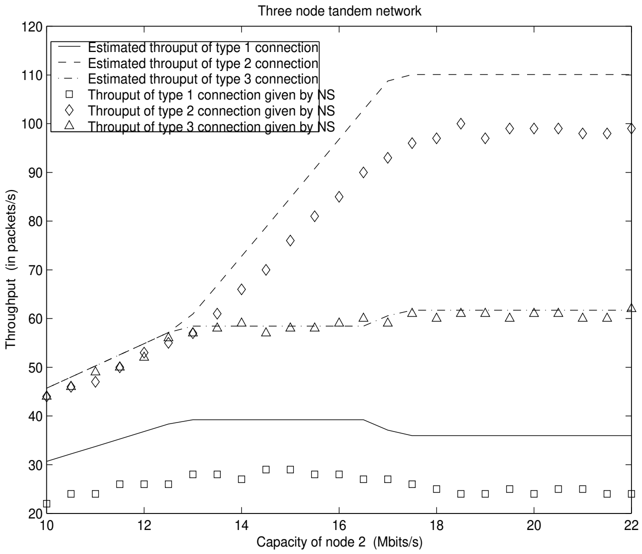

Fig. 9. Three node tandem network

lead to efficient computational procedures and FP formulation helps us to prove the existence of a solution. The presented model does not require the pre-identification of bottleneck links and include the possibility of the source rate limitation. Even simple Benchmark network examples demonstrate that the localization of bottlenecks is not intuitive and TCP throughput is not always a monotonous function of the total capacity of the network.

V. Appendix: Another proof of existence

Here we give another proof of existence of a solution to (4), (5) and (6), which is based on the Nash Theorem [6] and uses the technique proposed in [12]. Unfortunately, it is possible to use this approach only in the case of a simple relation between the throughput T_i and the packet loss probability on the path κ_i, such as square root formula. We chose to present both proofs since obtaining the existence of fixed point solutions are encountered often in other problems and one or the other proof techniques could thus be used in other networking contexts (such as in the setting of [3]).

Theorem 2: Let the relation between the throughput T_i and the packet loss probability on the path be given by the square root formula (2). Then, the system of (4), (5) and (6) has a solution.

Proof: Let us define the functions

$$f_v(p_v, p^v) := \mu_v - \sum_{i \in I} \gamma_{iv} \left(\prod_{u \in \pi_i(v)} (1 - p_u) \right) \times$$

$$N_i MSS_i \min\{\frac{1}{\theta_i}\sqrt{\frac{c_i}{\kappa_i(\mathbf{p})}}, \frac{W^i_{max}}{\theta_i}\}$$

and

$$h_v(p_v, p^v) := -(f_v(p_v, p^v))^2,$$

where $p^v = (p_1, ..., p_{v-1}, p_{v+1}, ..., p_{|V|})$. We have introduce the notation p^v, as we want to study the behavior of functions f_v and h_v with respect to the probability p_v, having the other probabilities fixed.

Next let us show that the functions h_v are quasi-concave in p_v, that is, the level sets $\{p_v | h_v(p_v, p^v) \geq a\}$ are convex. We note that the function $f_v(p_v, p^v)$ is a constant minus the sum of functions of the form

$$\varphi(p_v) = c(1 - p_v) \min\{\frac{1}{\sqrt{a + bp_v}}, d\},$$

where the constants $a, b \in [0, 1]$ and c depend on p^v. We note that the function $\varphi(p_v)$ is piece-wise differentiable on $[0, 1]$. In particular, we have

$$\varphi'(p_v) = \begin{cases} -cd, & \text{if } \dfrac{1}{\sqrt{a + bp}} > d, \\ -c\dfrac{(a + bp_v) + 0.5(1 - p_v)}{(a + bp_v)^{3/2}}, & \text{if } \dfrac{1}{\sqrt{a + bp}} < d. \end{cases}$$

Hence, we can see that this function is decreasing on the interval $[0, 1]$. Since the sum of decreasing functions is again decreasing, the function $f_v(p_v, p^v)$ is increasing in p_v. Now we consider two cases: (a) $f_v(0, p^v) \geq 0$ and (b) $f_v(0, p^v) < 0$. In the case (a), the function $h_v(p_v, p^v)$ is decreasing in p_v for the whole interval $[0, 1]$, and hence it is quasi-concave. Note that in the case (a) the function $h_v(p_v, p^v)$ achieves its maximum at $p_v = 0$. In the case (b), as $f_v(1, p^v) = \mu_i > 0$, the function $f_v(p_v, p^v)$ necessarily crosses zero on the interval $[0, 1]$. The latter implies that in the case (b) the function $h_v(p_v, p^v)$ is unimodal, and hence, quasi-convex. Moreover, in this case its maximal value is equal to zero.

Since all functions $h_v(p_v, p^v)$ are quasi-concave in p_v for any fixed p^v, we conclude from the Nash theorem [6] that there exists at least one set $(p_1^*, ..., p_{|V|}^*) \in \times_{v \in V}[0, 1]$ such that

$$p_v^* = \arg \max_{p_v \in [0,1]} h_v(p_1^*, ..., p_{v-1}^*, p_v, p_{v+1}^*, ..., p_{|V|}^*), \; v \in V.$$

¿From the proof of the quasi-concavity of $h_v(p_v, p^v)$, it immediately follows that either $p_v^* = 0$ or $f_v(p_v^*, p^{v*}) = 0$, and in both cases $f_v(p_v^*, p^{v*}) \geq 0$. Hence the set $(p_1^*, ..., p_{|V|}^*)$ is a solution to (4) and (5). ■

References

[1] E. Altman, K. Avrachenkov, and C. Barakat, "A stochastic model of TCP/IP with stationary random losses", *ACM SIGCOMM'2000*, Stockholm, pp.231-242.

[2] T. Bonald and L. Massoulié, "Impact of fairness on Internet performance", *ACM SIGMETRICS 2001*, pp.82-91.

[3] T. Bu and D. Towsley, "Fixed point approximation for TCP behaviour in an AQM network", *ACM SIGMETRICS'01*, June 2001.

[4] Claudio Casetti and Michela Meo, "A new approach to model the stationary behavior of TCP connections", *IEEE INFOCOM'2000*, March 2000.

[5] R.W. Cottle, J.-S. Pang, and R.E. Stone, *The linear complementarity problem*, Academic Press, Boston, 1992.

[6] V. Istratescu, *Fixed point theory*, Dordrecht, Holland: Reidel, 1981.

[7] F.P. Kelly, A. Maulloo, and D. Tan, "Rate control for communication networks: shadow prices, proportional fairness and stability", J. Oper. Res. Soc., v.49, 1998, pp.237-252.

[8] S. Kunniyur and R. Srikant, "End-to-end congestion control schemes: Utility functions, Random losses, ECN marks", *IEEE INFOCOM 2000*.

[9] S.H. Low and D.E. Lapsley, "Optimization flow control I: Basic Algorithm and Convergence", *IEEE/ACM Trans. on Networking*, v.7, no.6, Dec. 1999.

[10] L. Massoulié and J. Roberts, "Bandwidth sharing and admission control for elastic traffic", *Telecommunication Systems*, v.15, pp.185-201, 2000.

[11] V. Mistra, W.-B. Gong, and D. Towsley, "Fluid-based Analysis of a Network of AQM Routers Supporting TCP Flows with Application to RED", *ACM SIGCOMM*, Aug 2000.

[12] J. Mo and J. Walrand, "Fair end-to-end window-based congestion control", *IEEE/ACM Trans. Networking*, v.8, no.5, Oct 2000.

[13] J. Padhye, V. Firoiu, D. Towsley, and J. Kurose, "Modeling TCP Throughput: a Simple Model and its Empirical Validation", *ACM SIGCOMM*, Sep 1998.

[14] S. Floyd and V. Jacobson, "Random Early Detection gateways for Congestion Avoidance", *IEEE/ACM Transactions on Networking*, Aug 1993.

[15] R. Gibbens and P. Key, "Distributed control and resource pricing", *ACM SIGCOMM'2000 Tutorial*.

[16] V. Jacobson, "Congestion avoidance and control", *ACM SIGCOMM*, Aug 1988.

[17] NS-2, Network Simulator (ver.2), LBL, http://www-mash.cs.berkeley.edu/ns.

[18] M. Mathis, J. Semke, J. Mahdavi, T. Ott, "The Macroscopic Behaviour of the TCP Congestion Avoidance Algorithm", *ACM Computer Communication Review*, 27(3), July 1997.

[19] J.M. Ortega and W.C. Rheinboldt, *Iterative solution of nonlinear equations in several variables*, Academic Press, 1970.

[20] T. Ott, J. Kemperman, and M. Mathis, "The stationary behavior of the ideal TCP congestion avoidance", ftp://ftp.telcordia.com/pub/tjo/TCPwindow.ps

[21] V. Paxson, "End-to-End Routing Behavior in the Internet", *ACM SIGCOMM*, Aug 1996.

[22] M. Vojnović, J.-Y. Le Boudec, and C. Boutremans, "Global fairness of additive-increase and multiplicative-decrease with heterogeneous round-trip times", *IEEE INFOCOM 2000*.