Using TCP Models To Understand Bandwidth Assurance in a Differentiated Services Network

M Baines
Carleton University
Ottawa, Canada

N Seddigh, B Nandy, P Pieda
Tropic Networks
Ottawa, Canada

M Devetsikiotis
NC State University
Raleigh, North Carolina

***Abstract*—In this paper, a comprehensive analytical model to predict the bandwidth achieved by aggregates of TCP flows in a Diffserv network is presented. The model predicts achieved bandwidth in three different cases: an over-provisioned network, an under-provisioned network, and a near-provisioned network. In developing the model, we ensure that all parameters are measurable using standard tools and information available from routers and network management tools in today's networks.**

Simulation was used to establish the validity of the model and understand its scope of applicability and limitations. Using the model, we explain why achieved excess bandwidth is based on factors such as RTT, packet size, and CIR. Finally, we present a novel extension of the model to predict the bandwidth of TCP flows in a Diffserv network with multiple congested nodes.

***Index terms*—Assured Forwarding PHB, Diffserv, multiple congested nodes, MRED, TCP Modelling, TCP Throughput**

1 INTRODUCTION

The IETF's DiffServ Working Group has defined two PHB groups: the Expedited Forwarding (EF) PHB and the Assured Forwarding (AF) PHB group. This paper focuses on the latter PHB. Typically, an ISP would use the AF PHB to provide a service where customer packets from a site are forwarded with high probability as long as the aggregate traffic from the site does not exceed a previously agreed upon Committed Information Rate (CIR). The site may exceed the CIR with the understanding that the excess traffic is not delivered with as high a probability as the traffic that is within the profile.

The AF PHB group is divided into four independently forwarded classes. Within each class, a packet is assigned to one of three different levels of drop precedence. However, during times of congestion, packets with higher drop precedence are dropped with higher probability, or forwarded with lower probability, than packets with lower drop precedence.

MRED-based (Multi-level Random Early Detection) [1] Active Queue Management (AQM) schemes are one way to provide Assured Forwarding. Though the AF PHB has become standardized, there are still some open issues that need to be understood. The most important question relates to the kind of end-to-end service that can be created using such schemes. There is concern that AF should show some measure of predictability and fairness of service for paying customers.

If subscribers were charged based on their CIR then, in a fair system, one would expect subscribers who have paid the same amount to obtain equal bandwidth. Unfortunately, this is not the case. Earlier work [2] [3] has shown that bandwidth is shared in a manner that is not totally dependent on the CIR. Achieved bandwidth is dependent on factors such as RTT (Round Trip Time), packet drop probability, packet size and the CIR. Therefore, it is possible for subscribers with the same CIR to obtain dramatically different shares of the excess, or uncommitted, bandwidth. Customers will be unhappy to know that another customer who has paid for the same CIR is getting significantly more bandwidth than they are.

Therefore, it is important to understand exactly how the CIR affects bandwidth distribution. In the literature there are many studies [4][5] which have proposed steady-state analytical throughput models of TCP in best-effort networks. This paper proposes to extend this previous work and factor the CIR into a new model, which looks at TCP throughput in an AF network.

There are three main contributions of this work: 1) We extend previous work in best-effort TCP modeling and develop a formula for the steady-state throughput of an aggregate of long-lived TCP flows that subscribe to a service based on the AF PHB. In this formula, we include the CIR as one of the key factors affecting throughput. 2) We use the formula to make key observations about how the individual factors affect the throughput. Some of these observations have already been presented in other papers [2] [3] (as observations of simulation results or test network results). Using the analytical model, we confirm and explain these previous observations. Such work can have a significant contribution to better understanding how bandwidth is distributed in an AF network and how TCP congestion avoidance causes unfair bandwidth distribution. 3) We present a novel extension of our work in which we examine the effect of multiple congested nodes on bandwidth assurance.

The organization of the paper is as follows: Section 2 reviews related work in this area. Section 3 and 4 provides the simple TCP model. Section 5 presents the key inferences and simulation results that are used for validation. Section 6 examines the case of multiple congested nodes. In Section 7 we present our discussion and Section 8 concludes the paper.

2 RELATED WORK

There have been a number of recent studies that proposed steady-state throughput models of TCP. Some of the earliest work was done in [4], where Mathis et al developed a steady-state throughput model of the saw-tooth behavior of the TCP Congestion Avoidance algorithm.

Under the assumption of constant RTT, low to moderate packet loss, and random constant drop probability, the congestion window (cwnd) follows a perfect saw-tooth. Periodically the congestion window rises to a maximum

0-7803-7208-5/01/$17.00 (C) 2001 IEEE

value at which time packet loss causes the window to be halved due to the fast retransmit mechanism. By counting the number of packets sent in one saw-tooth and measuring its period, Mathis et al were able to come up with the following equation for bandwidth of a single TCP flow:

$$BW = \frac{MSS}{RTT}\frac{C}{\sqrt{p}} \quad (1a)$$

where *MSS* is the maximum segment size, *C* is a constant which is dependant on the type of TCP, and *p* is the packet drop probability.

There are a number of situations where the model is not expected to apply. These include the case where the data receiver is announcing too small a window or the sender does not always have data too send. Also, the model does not account for timeouts or short connections that don't reach steady state.

In [5], Padhye et al develop a model which captures not only the behavior of TCP's fast retransmit mechanism but also the effect of TCP's timeout mechanism on throughput. Their model is more accurate than the one developed by Mathis, especially for the case of networks where there are many timeouts. However, in this paper we ignore the effect of timeouts.

In [6] and [7], the authors propose simple models of TCP behavior in a DiffServ networks. In [6], Sahu et al propose a single flow model for TCP with token-bucket marking. In [7], Yeom and Reddy propose a model for TCP with short term rate-based marking. Here, we propose a model for TCP aggregates with long term rate-based marking. Long term rate-based marking is known [8] to work well with flow aggregates while short-term rate based marking can perform better for a single flow.

3 Extending the Basic TCP Model for Aggregate of Flows

Because the AF PHB is concerned with aggregates of flows, we begin by extending equation (1a) to handle an aggregate of flows with the same source and destination by summing the throughput of the individual flows. The equation for an aggregate of flows is then:

$$BW = \sum_{i=1}^{NoF} \frac{MSS_i}{RTT_i}\frac{C}{\sqrt{p_i}}$$

where NoF = the number of flows in the aggregate, MSS_i is the average packet size for TCP flow i, RTT_i is the average round trip time for TCP flow i, and p_i is the packet drop probability for TCP flow i.

We can construct a simple model for an aggregate of flows by assuming that all flows have the same average round trip time, $RTT_i = RTT$ ($1 \le i \le NoF$), the same average packet size, $MSS_i = MSS$, and the same packet drop probability, $p_i = p$. (This is similar to the approach followed by Firoiu and Borden in [9].) We can now drop the index i and reduce the formula to the following:

$$BW = NoF * \frac{MSS}{RTT}\frac{C}{\sqrt{p}} \quad (1b)$$

We validated equation (1b) by running simulations where we increased the number of flows per aggregate for every successive simulation run. As the number of flows per aggregate increased, the difference between the achieved and predicted bandwidth remained low.

4 Simple TCP Model For DiffServ

A simple model is now presented which extends the work done in [4] by adding the CIR, as a factor in determining achieved bandwidth. Our model is based on the assumptions presented below (most of which are taken from [4]):

1. The receiver window is never reached.
2. Sender always has data to send.
3. Little or no TCP timeouts.
4. TCP Reno or TCP SACK.
5. Non-overlapping RED thresholds. The maximum threshold for dropping out-of-profile packets is less than the minimum threshold for dropping in-profile packets.

We now begin developing our new model by first looking at the over-provisioned case.

4.1 Over-provisioned Case

1. The network is over-provisioned when the cumulative sum of the CIRs of all of the aggregates using a link is less than the bandwidth of the link.
2. All packet drops are out-of-profile packets. There are no in-profile packet drops.

The probability of dropping an out-of-profile packet will be represented by the symbol p_{out}. The probability p_{out} can be related to *p* by the probability P_{out}, of marking a packet as out-of-profile. If we assume a long term rate-based [8] policer that marks CIR worth of packets in-profile over the long term, then packets should be marked out-of-profile with probability:

$$P_{out} = \frac{avgrate - CIR}{avgrate} = \frac{BW - CIR}{BW} \quad (2),$$

when the *avgrate* exceeds CIR . In the equation above, *avgrate* is a sliding window estimate of the bandwidth, BW. An expression for p in terms of p_{out} can now be developed.

$$p = P_{out}p_{out} = \frac{BW - CIR}{BW}p_{out} = p_{out} - p_{out}\frac{CIR}{BW} \quad (3)$$

Equation (1) can now be redeveloped in terms of pout. Replacing the p in equation (1) with the expression above results in the following equation:

$$BW = \frac{NoF * MSS}{RTT} \frac{C}{\sqrt{p_{out} - p_{out}\frac{CIR}{BW}}} \quad (4)$$

Solving the quadratic, and rejecting the negative root (BW>0):

$$BW = \frac{CIR}{2} + \sqrt{\left(\frac{CIR}{2}\right)^2 + \left(\frac{NoF * C * MSS}{\sqrt{p_{out}}RTT}\right)^2} \quad (5).$$

Note that equation (5) is equivalent to equation (1b) when CIR = 0. It can also be shown that the equation (5) is equivalent to the equations developed independently in [6] and [7] for token-bucket and short-term TSW marking respectively.

It is easy to derive an equation for the excess bandwidth (i.e. the bandwidth obtained beyond the CIR).

$$BW_{ex} = \sqrt{\left(\frac{CIR}{2}\right)^2 + \left(\frac{NoF * C * MSS}{\sqrt{p_{out}}RTT}\right)^2} - \frac{CIR}{2} \quad (6)$$

4.2 Near-provisioned case

1. The network is near-provisioned when the cumulative sum of the CIRs of all of the aggregates using a link is close to (within 10%) the bandwidth of the link.
2. Both in and out-of-profile packets are dropped.

One cannot always assume that there are no in-profile packets dropped. It is common to have both in and out-of-profile packets dropped when the network is near fully provisioned. At such a time, the queue size can fluctuate between the in-profile and out-of-profile random dropping zones.

Below are the probabilities of having a packet marked in or out-of-profile.

$$P_{in} = \frac{CIR}{BW}\ ,\ P_{out} = \frac{BW - CIR}{BW}$$

From this, it is possible to develop a relation between the p in equation (1b) and p_{in} and p_{out}, which correspond to the probabilities of dropping in and out-of-profile packets respectively.

$$p = P_{in}p_{in} + P_{out}p_{out} = \frac{CIR}{BW}p_{in} + \frac{BW - CIR}{BW}p_{out}$$

Plugging the p above into (1b) and solving for BW, it is possible to come with a new expression for bandwidth.

$$BW = \frac{p_{out} - p_{in}}{2p_{out}}CIR + \sqrt{\left(\frac{p_{out} - p_{in}}{2p_{out}}CIR\right)^2 + \left(\frac{NoF * C * MSS}{\sqrt{p_{out}}RTT}\right)^2} \quad (7)$$

Note that if we let p_{in} = 0, then equation (8) reduces to equation (5).

4.3 Under-provisioned Case

1. The network is under-provisioned when the cumulative sum of the CIRs of all of the aggregates using a link is greater than the bandwidth of the link.
2. All packet drops are in-profile packets.
3. There are no packets marked out-of-profile.

Next, we look at the case of where no packets are marked out-of-profile (i.e. the case where the BW < CIR). In an MRED implementation where there are non-overlapping thresholds (max_{th} and min_{th} from [1]), this case corresponds to the case where the network is under-provisioned. The probability of marking a packet in-profile is now 1 and p is simply equal to the probability of dropping in-profile packets.

Plugging the p_{in} for p into equation (1b) results in the following expression for bandwidth:

$$BW = \frac{NoF * C * MSS}{\sqrt{p_{in}}RTT} \quad (8)$$

In this section we have presented three formulas to predict achieved bandwidth in the under-provisioned, near-provisioned and over-provisioned cases. In Section 5, we use simulation to validate our work. The simulation work focuses on the over-provisioned case because any CIR guarantee will have to rely on an over-provisioned network.

5 Simulation And Observations

We can use the model to make certain key observations on how the four factors (packet size, number of flows, RTT, and CIR) affect the steady-state throughput. We validate these observations, by showing them to be true in simulation.

The simulation was performed using the ns-2 network simulator, along with the Nortel Networks Diffserv code, which is available in the latest ns code snapshot.

The topology, presented in Fig. 1, consists of two aggregates of TCP flows. One aggregate sends traffic from source-0 to sink-0 and the other aggregate of flows sends traffic from source-1 to sink-1. We used SACK TCP because it is able to remain in Congestion Avoidance at higher loss rates than other TCP variants. The constant C for our particular setup was 1.31 as suggested in [4].

Unless noted otherwise, an aggregate of 15 TCP flows was connected between each source and sink. The individual TCP flows were long-lived FTP applications which transmitted MSS sized (1000 byte) packets. RIO [8] was used as the AQM scheme and the RED parameters were {10,300,0.05} and {300,400,0.02} for out-of-profile and in-profile packets respectively. A Time Sliding Window (TSW) [8] policer was used to mark packets as either in or out-of-profile using a CIR of 5.0 Mbps per aggregate.

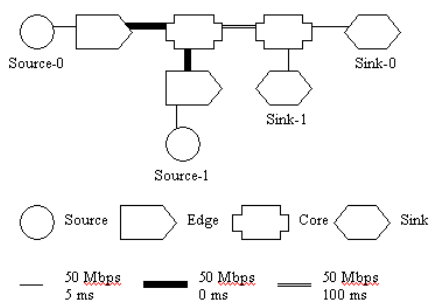

Fig. 1. Simulation Topology

5.1 Methodology

In each of the simulations that follow, we used formula (5) to calculate the predicted bandwidth. The formula requires that the *RTT*, *CIR*, *MSS*, drop probability of out-of-profile packets, p_{out}, and number of flows, *NoF*, be measured. In the simulations the first three factors were fixed beforehand. In a real network the CIR is controlled by network management, the number of flows and packet size can be approximated using average values.

The other two factors, p_{out} and *RTT*, were both calculated using simulation measurements. The drop probability was calculated by using measurements of out-of-profile packet drops and arrivals at the core node.

The average RTT was calculated using equation (9) below.

$$RTT = R_0 + \sum_{i=1}^{n} \frac{Q_{avgi}}{L_i} \quad (9)$$

where $Qavg_i$ is the average queue size at link i and L_i is the link speed at link i. The average RTT is the sum of the average waiting times in queues along the path and R_0, the propagation and transmission time on the rest of the round trip. In this particular topology the queue size is zero everywhere except at the first core node.

5.2 Observations

1st. All else being equal, an aggregate with a smaller CIR will obtain more excess bandwidth than an aggregate with a larger profile.

This can be demonstrated by showing that $\partial BW_{ex}/\partial CIR$ is always less than zero. If this partial derivative is less than zero, then a smaller CIR should result in more excess bandwidth, all else being equal.

This is a not always a desirable result. From a business perspective, a customer may expect that buying a larger profile would result in a larger share of the excess bandwidth and not a smaller share.

In the simulation graph presented in Fig. 2(a)., the CIR of aggregate 1 was increased with each simulation run and the bandwidth measured. The CIR of aggregate 0 was kept constant at 5.0 Mbps.

It should be easy to see that aggregate 0 gets more of the excess bandwidth (difference between the bandwidth and the CIR) even though it has a smaller CIR. In the very last point on the graph, aggregate 0 obtains around 17 Mbps of bandwidth resulting in 12 Mbps of excess bandwidth while aggregate 1 obtains 32 Mbps of bandwidth, resulting in only 7 Mbps of excess bandwidth.

2nd. All else being equal, an aggregate with a larger RTT will obtain less excess bandwidth than an aggregate with a smaller RTT.

By careful examination of equation (6), this observation should become obvious. In the simulation graph presented in Fig. 2(b)., the core-core link delay was reduced to 25ms. The source-edge and edge-sink delays were increased to 12.5ms for aggregate 0. The RTT of aggregate 1 was increased with each simulation run through its source-edge and sink-edge delays. As demonstrated in Fig. 2(b), our analytic model confirms earlier results shown in [2][3].

3rd. All else being equal, an aggregate with a larger number of flows will get more excess bandwidth than an aggregate with fewer flows.

By careful examination of equation (6), this observation should become obvious. Also, the same conclusion can be derived from the simulation graph presented in Fig. 2(c). In the simulation the number of flows in aggregate 1 was increased with simulation run. This same result has been shown in [2].

4th. All else being equal, an aggregate with a larger MSS (maximum segment size) will get more excess bandwidth than an aggregate with a smaller MSS.

By careful examination of equation (6), this observation should become obvious. Also, the same conclusion can be derived from the simulation graph presented in Fig. 2(d). In the simulation the packet size of the aggregate 1 flows was increased with each simulation run. This result has been shown via experimentation in [2].

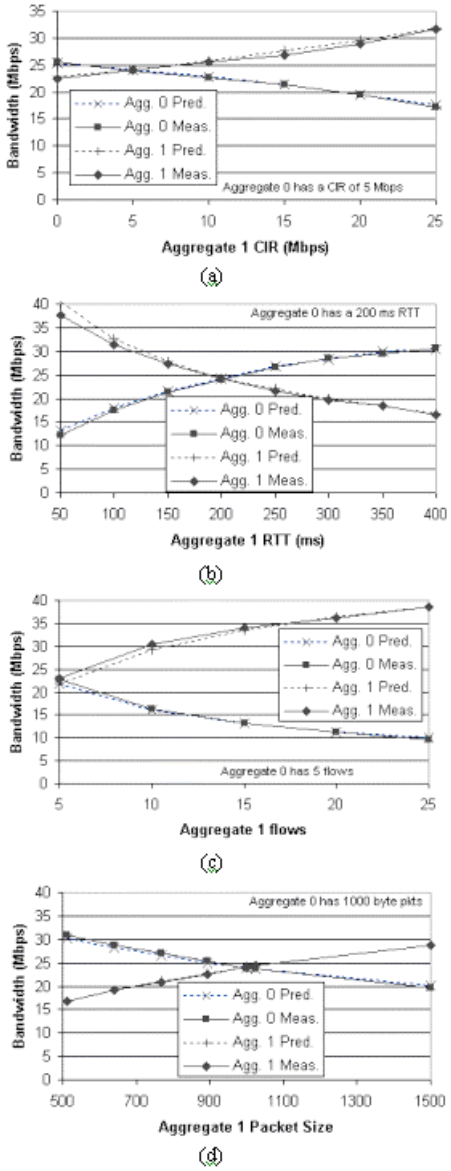

Fig. 3. Simulation Results

6 MULTIPLE CONGESTED NODES

In this section we investigate the effect of multiple congested nodes on achieved bandwidth. Letting $p_{out,i}$ represent the probability of dropping out-of-profile packets at a particular node i and p_{out} represent the total out-of-profile packet drop probability over the whole path, it is possbile to develop a formula for p_{out} in terms of the individual $p_{out,i}$.

For the case of a flow which travels through n congested nodes, the probability of not dropping an out-of-profile p_{out}, is related to the $p_{out,i}$ via the following formula:

$$p_{out} = 1 - \prod_{i=1}^{n} (1 - p_{out,i}) \quad (10)$$

Equation (10) and equation (5) can be used to present a fifth key observation.

5th. All else being equal, an aggregate going through more congested nodes will obtain less excess bandwidth than an aggregate going through fewer congested nodes.

From equation (5), it is possible to see that a larger value for p_{out}, results in less excess bandwidth. Then from equation (10), one can see that each congested node adds one term to the product thereby increasing p_{out}.

Equation (10) is validated in the simulation section that follows.

6.1 Simulation

The topology used in this section is similar to that used in the Section 5. A similar topology was used in [4] and [10] to observe the effects of multiple congested nodes. Because of the increase in the size of the simulation topology, many parameters from section 5 were reduced in value to allow the simulation to complete in a reasonable amount of time.

An aggregate of ten TCP flows was connected between each source i and sink i. The individual TCP flows were long-lived FTP applications which transmitted MSS (in this simulation 1000 bytes) size packets. The RED parameters were {10,80,0.1} and {80,150,0.02} for out-of-profile and in-profile packets respectively.

The CIR for the long (the aggregate of flows going from sink-0 to source-0) and short aggregates (the aggregates of flows going from source-1 to sink-1, .., source-n to sink-n) was kept the same, and four different values were tried: 2.5 Mbps, 5.0 Mbps, 7.5 Mbps, and 10.0 Mbps. Each CIR value corresponds to a different level of provisioning ranging from 20% to 80% of the bottleneck bandwidths.

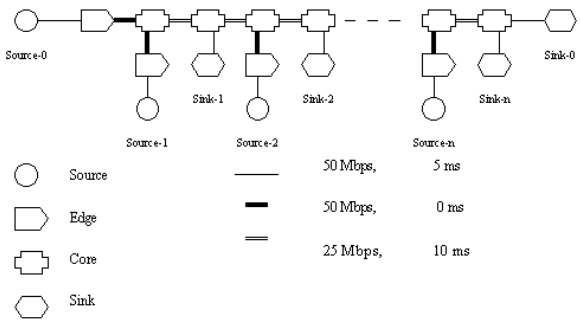

Fig. 3. Multiple Congested Nodes

In each successive simulation, the number of congested links was varied. The drop probability and queue size were measured for each congested node. These two values were used to calculate the RTT and the end-to-end out-of-profile drop rate for short and long aggregates. Also, the steady-state throughput obtained by the long aggregate of TCP flows was recorded.

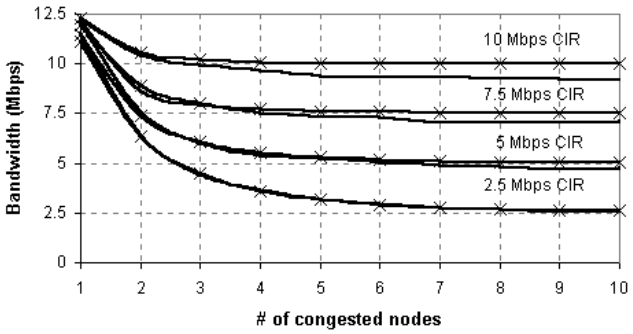

Fig. 4. Bandwidth vs. # of nodes for TCP SACK

Fig. 4 shows graphs of the measured and predicted throughput for the long source (source-0 to sink-0) TCP aggregates. The solid-line curves are the measured bandwidth and the X-line curves are the predicted bandwidth. Each curve graphs the bandwidth versus the number of congested nodes along the path. Each pair of predicted and measured curves approaches its CIR as the number of congested links increases.

The bandwidth decreases with the number of congested nodes and eventually stabilizes at the CIR. This can be explained through equation (5) presented in Section 4. As the number of congested nodes increases so do p_{out} and the RTT, decreasing the value of the second term of the square root. Eventually, this term is negligible compared to the left-hand term $(CIR/2)^2$. The bandwidth formula then reduces to BW = CIR.

At higher provisioning levels, the measured bandwidth seems to diverge from CIR when the number of congested nodes is high. We believe this is due to the fact that the average queue size is very large when the provisioning level is high. A high average queue size results in a higher probability of dropping packets and hence, a higher probability and consecutive packet drops resulting in timeouts. Also, at higher provisioning levels, in-profile packet drops can no longer be ignored.

It is also important to note that as the rate of provisioning goes up, the aggregate obtains less excess bandwidth. This is because there is less excess bandwidth to be distributed among aggregates.

7 DISCUSSION

This paper developed an analytical model to predict the bandwidth of TCP aggregates in an AF-based DiffServ network. The model covers the cases of under-provisioned, near-provisioned and over-provisioned networks. Extensive simulations were performed to: (i) validate the model, and (ii) study the impact of various factors (RTT, packet size, CIR, # of flows etc) on bandwidth assurance. There is also new work to understand the effect of multiple congested nodes on bandwidth assurance in a diffserv network.

The model is beneficial in expanding current understanding of how the five factors (RTT, packet size, CIR, number of flows in an aggregate, and number of congested nodes along the path) affect AF achieved bandwidth. We believe our model and its various extensions could be integrated as part of a network planning/engineering tool. The ability to predict the achieved bandwidth of TCP aggregates will be of use to network engineers when they layout and provision their networks.

The TCP models developed in this paper have a number of limitations as far as applicability is concerned. Firstly, the model is not applicable for aggregates consisting of short TCP flows (mice). Secondly, it assumes that the long-lived flows are operating consistently in the congestion avoidance region - i.e. no timeouts.

8 CONCLUSION

The main contribution of this paper is the development of an analytical model to predict the achieved bandwidth of TCP aggregates in a Differentiated Services Network. Further, the model covers the case of (i) over-provisioned, (ii) under-provisioned and (iii) near-provisioned networks. Another contribution of this paper is to extend the model for a network with multiple congested nodes. Simulation results validate the bandwidth predictions made using the model.

REFERENCES

[1] Makkar R, Lambadaris I, Salim J H, Seddigh N, Nandy B, and Babiarz J, "Empirical Study of Buffer Management Schemes for DiffServ Assured Forwarding PHB," *Proceedings of 9th International Conference on Computer Communications and Networks*, Las Vegas, October 2000.

[2] Seddigh N, Nandy B, and Pieda P, "Bandwidth Assurance Issues for TCP flows in a Differentiated Services Network", *Proceedings of Globecom '99*, Rio De Janeiro, December 1999.

[3] Yeom I, and Reddy Y, "Realizing throughput guarantees in a differentiated services network," Accepted at *IEEE Int. Conf. On Multimedia Computing and Systems*, June 1999.

[4] Mathis M, Semke J, Mahdavi J, and Ott J, "The macroscopic behaviour of the TCP congestion avoidance algorithm", *Computer Communication Review*, 27(3), July 1997.

[5] Padhye J, Firoiu V, Townsley D, and Kurose J, "Modeling TCP Throughput: A Simple Model and its Empirical Validation", CMPSCI Technical Report TR 98-008, University of Massachussetts, MA, 1999.

[6] Sahu S, Nain P, Towsley D, Diot C, and Firoiu V, "On Achievable Service Differentiation with Token Bucket Marking for TCP," in Proceedings of ACM SIGMETRICS'00, Santa Clara, CA, June 2000.

[7] Yeom I, and Reddy A, "Modeling TCP Behavior in a Differentiated Services Network," TAMU ECE Technical Report, May 1999

[8] Clark D, Fang W, "Explicit Allocation of Best Effort Packet Delivery Service", IEEE/ACM Trans. Networking, vol6, Aug. 1998.

[9] Firoiu V, Borden M, "A Study of Active Queue Management for Congestion Control", In Proceedings of INFOCOM 2000, March 2000.

[10] Floyd S, Connections with Multiple Congested Gateways in Packet-Switched Networks, Part1: One Way Traffic, *Computer Communications Review*, 21(5), October 1991.