Sizing Router Buffers for Large-Scale TCP/IP Networks

Hiroyuki Hisamatsu
Department of Computer Science
Osaka Electro–Communication University
1130–70 Kiyotaki, Shijonawate
Osaka, 575–0063 Japan
E-mail: hiroyuki.hisamatsu@olnr.org

Go Hasegawa
Cybermedia Center
Osaka University
1–32 Machikaneyama, Toyonaka
Osaka, 560–0043 Japan
E-mail: hasegawa@cmc.osaka-u.ac.jp

Masayuki Murata
Graduate School of Information Science
and Technology, Osaka University
1–5 Yamadaoka, Suita
Osaka, 565–0871 Japan
E-mail: murata@ist.osaka-u.ac.jp

Abstract— **We investigate the validity of reducing router buffer size in a large-scale network that includes both core networks and edge networks. We first devise a novel mathematical analysis method of estimating the average behavior of TCP connections in a network with 100/1,000/10,000 routers/endhosts/links and 100,000 concurrent TCP connections. By applying our analysis method to the Abilene-inspired network, we demonstrate the influence of small buffer on link utilization, packet loss ratio and the performance of TCP connections passing through the router. One important result is that, especially when the edge network becomes faster, decreasing buffer size at core routers causes unfairness between TCP connections that traverse the core network and those do not traverse the core network.**

keywords: Buffer Size, TCP (Transmission Control Protocol), Fluid Flow Approximation, Large-Scale Network

I. Introduction

The router performance is an important factor in Internet performance. The bandwidth of an output link and the buffer size of a router have a lot of influence on the TCP throughput passing through the router. Traditionally, the size of router buffers is determined by the product of the link bandwidth and the average round-trip time (RTT) of TCP connections passing through the router. However, recent studies [1, 2] have claimed that when the number of TCP connections is sufficiently large and the TCP connections are desynchronized, the buffer size can be decreased to the bandwidth-delay product divided by the square-root of the number of flows without the under utilization of the link bandwidth. In [3], authors consider the buffer size and the stability of a network by using control theory, and recommend that buffer size should be 50 packets or less. In [4], authors show that the link utilization can be kept sufficiently high with the buffer size of a core router at 20 packets when pacing TCP [4] is deployed or the access link is much slower than the backbone link. Using the ns-2 simulator, in [5], authors have shown that when router buffer size is determined by [1], the packet loss ratio increases by about 5% to 15% and there is a large variations in the TCP throughput in the network.

However, these studies [1-4] focus almost exclusively on the link utilization, and the throughput of TCP connections passing through the router is not taken into consideration. That is, the increase in packet loss ratio due to decreased buffer size and its influence on TCP performance are not considered. Although the authors focus on the TCP throughput in [5], the network topology used in this study is quite simple, with only about 400 TCP connections passing through the bottleneck link, whereas 500–1,000 concurrent TCP connections are required to avoid the synchronization [1]. Therefore, the effect of the small buffer size is verified only in the small-scale networks that focus only on the core network, and the relationships between the core network and the edge network are not considered. One of the reasons for this may be that network simulators or analysis methods are limited. These studies also explicitly or implicitly assumes that edge networks are slower than core networks. The impact of the small buffer size on network performances in today's networks, where the access link speed has been increasing, should be investigated.

In this paper, we investigate the validity of reducing buffer sizes in a large-scale network that includes both core networks and edge networks. To do this, we first devise a novel method of analyzing the average behavior of TCP connections in the large-scale networks with over 100/1,000/10,000 routers/endhosts/links and 100,000 concurrent TCP connections. In our analysis, we model each network component (endhost's TCP and network link with Tail-Drop buffer) as an independent system, and then combine them into one system in order to analyze the average behavior of the TCP connections for the entire network. We apply the proposed analysis method to the Abilene-inspired network [6], which is developed based on the characteristics of the actual router-level topology. By deriving the link utilization and the packet loss ratio in each link, the TCP throughput and the TCP round-trip time in the network, we dispute the influence of the small buffer size on the whole network and on TCP connections. One of the important results of our analysis is that the fairness among TCP connections that pass through the core network and those that do not becomes worse than the well-known TCP unfairness caused by different round-trip times.

This paper is organized as follows. In Section II, we introduce the network and traffic models used in this paper. In Section III, we describe the analysis methods used in

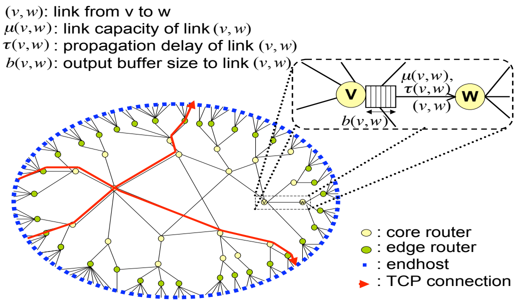
(v, w): link from v to w
μ(v, w): link capacity of link (v, w)
τ(v, w) : propagation delay of link (v, w)
b(v, w): output buffer size to link (v, w)
V
W
μ(v, w),
τ(v, w)
(v, w)
b(v, w)
○ : core router
● : edge router
▪ : endhost
► : TCP connection

Fig. 1: Network Model in the Analysis

modeling a TCP and a network link as independent systems. By combining these models, we obtain the model for the entire network. In Section IV, by presenting several numerical examples, we dispute the impact of the small buffer size on the core network. Finally, in Section V, we conclude the present paper and discuss future work.

II. Network and Traffic Models

A. Network Model

Fig. 1 shows the network model used in the analysis. The model consists of nodes and links. Each node corresponds to a endhost or a router. Let v and w ($v, w \in \mathcal{R}$) be nodes, where $\mathcal{R}$ is a set of nodes in a network. The ordered pair (v, w) refers to the unidirectional link from node v to node w. Note that, in this analysis, the link (v, w) differs from the link (w, v). Let $\mathcal{L}$ be a set of links in a network and $\mathcal{L}(\chi)$ be a set of links that the TCP connection χ traverses. The link capacity and the propagation delay of link (v, w) are denoted by $\mu_{(v,w)}$ and $\tau_{(v,w)}$, respectively. Each router is assumed to have separate output buffers for each outgoing link. The buffer size of the output link buffer to link (v, w) at node v is denoted by $b_{(v,w)}$.

TCP connections are established between endhosts according to the amount of offered traffic defined in Section II-B. $\mathcal{C}$ is a set of all TCP connections in the network. After determining the route that each TCP connection traverses, we can determine $\mathcal{C}(v, w)$, which is a set of TCP connections that traverse link (v, w). In the numerical example in Section IV we use Dijkstra's shortest path algorithm for determining the route that each TCP connection traverses. Note that we could apply any kind of routing algorithm in this analysis. For example, we could evaluate the effect of the overlay routing algorithm by applying the routing algorithm for the TCP connections that join the overlay network.

In this paper, we use a Drop-Tail discipline at an output buffer of each network, and focus on the average behavior of queue occupancy at the buffer. Note that we can apply other kinds of queuing disciplines, such as Random Early Detection (RED) and a mixture of multiple disciplines, by applying appropriate models to router buffers. For example, for RED discipline, we can use the existing model in [8].

B. Traffic Model

The amount of network traffic is determined using the gravity model [9]. By applying the basic gravity model, we assume that the amount of traffic from router v to router w is proportional to the product of the amount of traffic that enters the network at router v and the amount of traffic that leaves the network at router w. In this analysis, we assume that the network traffic is generated from the edge router in Fig. 1 to that the endhost is connected. We also assume that the amount of traffic injected into/leaving from the edge router is proportional to the number of endhosts connected to the edge router. Then, the number of TCP connections between edge routers is determined to be proportional to the amount of traffic between the edge routers. In summary, the number of TCP connections that traverse from v to w is defined as:

$$N_{(v,w)} \quad = \quad \lfloor \alpha \times E_v \cdot E_w \rfloor, \tag{1}$$

where E_v and E_w are the numbers of endhosts connected to the edge routers v and w, respectively, and α is a weight parameter for determining the overall amount of network traffic.

In this paper, for the sake of simplicity, we employ the TCP Reno version for TCP traffic. Note that we can easily treat other versions of TCP by using appropriate models for TCP throughput of those version. Moreover, we can also analyze a network that has different TCP versions in the same network. Hereafter, TCP Reno is simply denoted as TCP unless noted otherwise.

III. Analysis

In the analysis, we first model a TCP and a network link as independent discrete-time systems with a Δ time slot. We then combine them into an entire network system and create simultaneous equations. By solving the equations, we can derive various network characteristics, such as the window size and throughput of TCP connections, the buffer occupancy and the packet loss ratio of network links. We also propose a method of decreasing the complexity of the simultaneous equations by removing links that do not cause congestion. We omit the explanation of the complexity reducing method due to the space limitation.

A. Models of TCP Behavior

We focus on the average behavior of a TCP connection, which is affected by the average window size depending on the packet loss ratio and round-trip time (RTT). That is, we model a TCP connection as a system with two inputs (packet loss ratio and number of packets in the output buffer of each network link) and one output (congestion window size of the TCP connection). The packet loss ratio of the TCP connection is denoted by d, and the TCP congestion window size is denoted by w. Change in the TCP congestion window size is given by [10]

$$\begin{aligned} w \;\leftarrow\; & w + (1-d)\frac{1}{w} - d\,(1-d_{TO}(w,d))\,\frac{1}{2}\frac{4\,w}{3} \\ & -d\,d_{TO}(w,d)\left(\frac{4\,w(k)}{3}-1\right), \end{aligned}$$

where $d_{TO}(w,d)$ is the probability that TCP detects packet loss by the timeout mechanism when the window size is w and the packet loss probability is d [11]:

$$d_{TO}(w,d) = \frac{(1-(1-d)^3)\,(1+(1-d)^3\,(1-(1-d)^{w-3}))}{(1-(1-d)^w)}.$$

Given a packet loss ratio $d_\chi(k)$ and a RTT $r_\chi(k)$ at slot k, the congestion window size of the TCP connection χ at slot $k+1$, $w_\chi(k+1)$, can be calculated using the following equations [10];

$$\begin{aligned} w_\chi(k+1) = w_\chi(k) + \frac{w_\chi(k)}{r_\chi(k)}\,\Delta\Big(& \big(1-d_\chi(k)\big)\frac{1}{w_\chi(k)} \\ & -d_\chi(k)\Big(1-d_{TO}\big(w_\chi(k),d_\chi(k)\big)\Big)\,\frac{1}{2}\frac{4\,w_\chi(k)}{3} \\ & -d_\chi(k)\ d_{TO}\big(w_\chi(k),d_\chi(k)\big)\Big(\frac{4\,w_\chi(k)}{3}-1\Big)\Big). \end{aligned} \quad (2)$$

Let $q_{(v,w)}(k)$ and $d_{(v,w)}(k)$ be the number of packets in the output buffer and the packet loss ratio at link (v,w) at slot k, respectively. We can derive the packet loss ratio for TCP connection χ at slot k, denoted by $d_\chi(k)$, as follows;

$$d_\chi(k) \;=\; 1 - \prod_{(v,w)\in\mathcal{L}(\chi)} (1-d_{(v,w)}(k)). \quad (3)$$

Note that $\mathcal{L}(\chi)$ is a set of links that TCP connection χ traverses. We can also derive the RTT $r_\chi(k)$ of the TCP connection χ as follows;

$$r_\chi(k) \;=\; \sum_{(v,w)\in\mathcal{L}(\chi)}\left(\tau_{(v,w)} + \frac{q_{(v,w)}(k)}{\mu_{(v,w)}}\right), \quad (4)$$

where $\tau_{(v,w)}$ is the propagation delay of link (v,w).

B. Models of Network Link

We focus on the behavior of a network link when TCP connections, each of which has a certain value of congestion window size, traverse the link. Therefore, the network link is modeled as a system with one input (window sizes of TCP connections) and two outputs (packet loss ratio and number of packets in the output buffer of each network link).

The number of packets in the output buffer of link (v,w) at slot $k+1$, $q_{(v,w)}(k+1)$, is given by

$$\begin{aligned} & q_{(v,w)}(k+1) = \\ & \qquad q_{(v,w)}(k) + \Delta \sum_{\chi\in\mathcal{C}(v,w)} \big(\lambda_\chi(k) - \mu_{(v,w)}\big), \end{aligned} \quad (5)$$

where $\sum_{\chi\in\mathcal{C}(v,w)}\lambda_\chi(k)$ is the sum of the throughput of TCP connections traversing link (v,w) at slot k, and $\mu_{(v,w)}$ is capacity of the link (v,w). The throughput of the TCP connection $\lambda_\chi(k)$ is given by the following equation from the congestion window size and the RTT of TCP connection, given by Eqs. (2) and (4).

$$\lambda_\chi(k) \;=\; \frac{w_\chi(k)}{r_\chi(k)} \quad (6)$$

Authors [1] have revealed the following characteristic of TCP connections traversing a link: when the number of TCP connections is sufficiently large and the TCP connections do not behave in a synchronized fashion, the sum of the congestion window size of the TCP connections follows a normal distribution. Since we are interested in large-scale networks with a large number of TCP connections, we can utilize this characteristic. Then, we can calculate $d_{(v,w)}(k)$, the packet loss ratio at the buffer of link (v,w) at slot k, as follows [1];

$$\begin{aligned} d_{(v,w)}(k) &= Prob[q_{(v,w)}(k) > b_{(v,w)}] \\ &= 1 - \frac{1}{2}Erf\left(\frac{b_{(v,w)} - q_{(v,w)}(k)}{\sigma(q_{(v,w)}(k))}\right), \end{aligned} \quad (7)$$

where $\sigma(q_{(v,w)}(k))$ is the standard deviation of the number of packets in the output buffer of link (v,w) at slot k, and $Erf()$ is the error function. Ignoring the number of dropped packets, the standard deviation of the number of packets in the output buffer of link (v,w) can be identical to that of the sum of the congestion window size of the TCP connections traversing link (v,w) [1];

$$\sigma(q_{(v,w)}(k)) = \sigma\left(\sum_{\chi\in\mathcal{C}(v,w)} w_\chi(k)\right). \quad (8)$$

We therefore derive the standard deviation of the sum of the congestion window size of the TCP connections for deriving $\sigma(q_{(v,w)}(k))$.

By assuming that the TCP connection is always in the congestion avoidance phase (this assumption is reasonable when the packet loss ratio is small), we can regard the variation of the congestion window size as a uniform distribution with a lower limit of $2\overline{w}/3$ and an upper limit of $4\overline{w}/3$, where $\overline{w}$ is the average size of the congestion window of the TCP connection. Consequently, we can obtain the standard deviation of the window size of the TCP connection as follows;

$$\sigma(w) \;=\; \frac{\overline{w}}{3\sqrt{3}}.$$

By assuming that the window size of all TCP connections follow an identical independent distribution, we can determine the standard deviation of the sum of the window size of the TCP connections traversing link (v,w) by using the following equation;

$$\sigma\left(\sum_{\chi\in\mathcal{C}(l_{v,w})} w_\chi(k)\right) \;=\; \sigma\left(\sqrt{\sum_{\chi\in\mathcal{C}(v,w)} \sigma(w_\chi(k))^2}\right) (9)$$

TABLE I

LINK PROPERTY

Link	Bandwidth	Prop. Delay
l_{cc}	10 [Gbit/s] (OC192)	0.01 [ms]
l_{cm}	10 [Gbit/s] (OC192)	0.1 [ms]
l_{me}	10 [Gbit/s] (OC192)	0.1 [ms]
l_{ee}	1 [Gbit/s] (GE)	1 [ms]

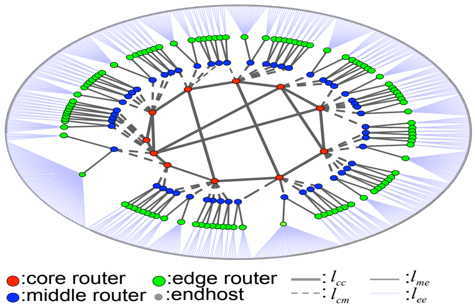

Fig. 2: Abilene-inspired Network [6]

C. Connecting Systems and Analysis

The congestion window size, RTT, packet loss ratio, throughput of TCP connection χ, packet loss ratio at output buffer of link (v, w) and number of packets in output buffer of link (v, w) in steady sate ($k \to \infty$) are denoted by w^*_χ, r^*_χ, d^*_χ, λ^*_χ, $d^*_{(v,w)}$, and $q^*_{(v,w)}$ respectively. We can regard Eqs. ((2) – (9)) as simultaneous equations by equating $w_\chi(k+1) \equiv w_\chi(k) \equiv w^*_\chi$ and $q_{(v,w)}(k+1) \equiv q_{(v,w)}(k) \equiv q^*_{(v,w)}$. By equating $w_\chi(k+1) \equiv w_\chi(k) \equiv w^*_\chi$ and $q_{(v,w)}(k+1) \equiv q_{(v,w)}(k) \equiv q^*_{(v,w)}$, Δ is eliminated in the simultaneous equations. That is, the solutions of simultaneous equations and Δ are independent of each other. By solving these simultaneous equations, we can derive the window size of each TCP connection, the number of packets in each output buffer and the packet loss ratio at each network link. The straightforward design of the analysis is one of the advantages of our analysis method.

IV. EFFECT OF DECREASING ROUTER BUFFER SIZE

In this section, we investigate the effect of decreasing buffer size on the performance of the whole network and of TCP connections by applying the proposed analysis method to the Abilene-inspired network [6]. Due to limitation of space, we do not show the validation of our analysis method, we verified the validity of our analysis method by comparing the ns-2 simulation results with analysis ones. Fig. 2 shows the Abilene-inspired network used in this section. This network topology is designed based on the characteristics of the actual router-level topology where the core routers have a smaller number of links with higher bandwidth, whereas the edge routers have a larger number of links with lower bandwidth. The topology consists of 11 core routers, 54 middle routers, 106 edge routers and 812 endhosts. There are 901 bidirectional links between routers and endhosts. For simplicity, we denote links between core routers as l_{cc}, those between core routers and middle routers as l_{cm}, those between middle routers and edge routers as l_{me}, and those between edge routers and endhosts as l_{ee}.

We compare the performance of the network and TCP connections when the small buffer is deployed in the core network with performance when the traditional size buffer is deployed in the core network. We determine the traditional buffer size of the output buffer of l_{cc} by the bandwidth-delay product $C \times \overline{RTT}$, where C is the link bandwidth, and $\overline{RTT}$ is the average round-trip time of TCP connections. Small buffer size is determined by [1], which corresponds to the bandwidth-delay product divided by the square-root of the number of TCP connections $C \times \overline{RTT}/\sqrt{N}$, where N is the number of TCP connections. Note that the buffer size of the output buffer of l_{cm}, l_{me}, and l_{ee} is determined by the bandwidth-delay product.

We consider the following two situations for analysis: the first case is that the bandwidth of the edge network link is comparatively small, and the other case is that the bandwidth of the edge network links is large. On account of space, we just give the brief explanation for the result of the former situation. When the link bandwidth of edge network is small, we can not observe significant differences in performance between traditional and small buffer sizings. This is because the bottleneck in the network is located at l_{me} in this case. That is, the traffic injected into to l_{cc} is limited by the bandwidth of l_{me}. Thus, the utilization of l_{cc} is not so large to make much differences in the packet loss ratio of l_{cc}. In terms of the TCP throughput, we can conclude that decreasing the buffer size in the network where the edge network is slow makes no advantage.

We then investigate the effects of decreasing buffer size when the bandwidth of the edge network links increases. Tab. I shows the parameter settings for links. We set α in Eq. (1) to 0.12, which means the total number of TCP connections in the network becomes $52,394$. The average two-way propagation delay of TCP connections in the network is about 4.83 [ms] and the average number of TCP connections on the link l_{cc} is $3,000$, and we use these values to determine the buffer size. We expect that, with these settings, the link utilization of the core network (l_{cc}) would increase and the conditions in [1] for decreasing buffer sizes would be satisfied.

Tab. II shows the average throughput and the average round-trip time of TCP connections, the average link utilization, and the average packet loss ratio for traditional and small buffers. We also found that the link utilization and the packet loss ratio of l_{cc} with small buffers are smaller than those of l_{cc} with traditional size buffers. When the buffer size of the output buffer of l_{cc} decreases, the packet loss ratio of l_{cc} increases, which degrades the throughput of TCP connections passing through the core network (l_{cc}). Then the TCP connections that do not traverse the core network dispossess the TCP

TABLE II

PERFORMANCE OF NETWORK AND TCP CONNECTIONS WHEN EDGE NETWORK IS FAST

buffer	link	Link Utilization	Packet Loss Ratio	TCP Throughput [Mbit/s]	Round-Trip Time [ms]
small	l_{cc}	0.638	0.001973	9.29	16.72
	l_{cm}	0.737	0.000077		
	l_{me}	0.459	0.000989		
	l_{ee}	0.690	0.000003		
traditional	l_{cc}	0.968	0.004195	9.23	23.93
	l_{cm}	0.748	0.000274		
	l_{me}	0.456	0.000665		
	l_{ee}	0.685	0.000006		

TABLE III

TCP THROUGHPUT [MBIT/S] BY NUMBER OF HOPS

buffer	4 hops	6 hops	7 hops	8 hops	9 hops	10 hops
small	160.32	67.33	3.29	2.41	2.15	1.96
traditional	141.94	47.04	7.10	3.33	2.32	1.79

connections that traverse the core network in the edge network (l_{cm} and l_{me}). As a result, the link utilization of l_{cc} and the packet loss ratio of l_{cc} with the small buffers are lower than those with traditional size buffers. In addition, there is almost no difference in the average TCP throughput in the network between small buffers and traditional size buffers. On the other hand, the average round-trip time of the traditional size buffers is larger than that of the small buffers. This is because traditional size buffers are much larger than small buffers and because the link utilization of l_{cc} is high.

Next, we investigate the TCP throughput in more detail. Tab. III shows the TCP throughput for the different numbers of link hops that TCP connections traverse. Note that the TCP connections with four and six hops do not traverse the core network and the TCP connections with seven or more hops do. The table shows that the throughput of the TCP connections which do not traverse the core network with small buffers is greater than that with traditional size buffers. On the other hand, the throughput of the TCP connections which traverse the core network with small buffers is less than that with traditional size buffers. This supports the explanation of low packet loss ratio and the small link utilization of l_{cc} when small buffers are deployed in the core network. That is, the TCP connections that do not traverse the core network dispossess the TCP connections that traverse the core network in the edge network (l_{cm} and l_{me}). This means that by decreasing the buffer size of the core routers, the relationship between TCP connections that pass through the core network and those do not is less fair than the well-known TCP unfairness caused by different round-trip time. From these results, we can conclude that the buffer size of the core routers should not be decreased when the utilization of the core network links is sufficiently high.

V. CONCLUSION AND FUTURE WORK

In this paper, we reported the effect on the network and TCP connections of decreasing buffer size in the core network. We first proposed a novel method of analyzing the average behavior of TCP connections in large-scale networks with 100/1,000/10,000 routers/endhosts/links and 100,000 concurrent TCP connections. We investigated the effectiveness of decreasing the buffer size of the core routers at both high and low utilization of core network links by applying our analysis method to the Abilene-inspired network. We concluded that decreasing core router's buffer size has almost no merit when the performance of the whole network, including the core and the edge networks, is taken into account. Future work will further investigate decreasing buffer size, especially with regard to the mixture of network services that includes streaming services.

REFERENCES

[1] G. Appenzeller, I. Keslassy, and N. McKeown, "Sizing router buffers," in *Proceedings of ACM SIGCOMM*, Sept. 2004, pp. 281–292.

[2] D. Wischik and N. McKeown, "Part I: Buffer sizes for core routers," *ACM SIGCOMM Computer Communication Review*, vol. 35, no. 3, pp. 75–78, July 2005.

[3] G. Raina, D. Towsley, and D. Wischik, "Part II: Control theory for buffer sizing," *ACM SIGCOMM Computer Communication Review*, vol. 35, no. 3, pp. 79–82, July 2005.

[4] M. Enachescu, Y. Ganjali, A. Goel, N. McKeown, and T. Roughgarden, "Routers with very small buffers," in *Proceedings of IEEE INFOCOM 2006*, Apr. 2006.

[5] A. Dhamdhere and C. Dovrolis, "Open issues in router buffer sizing," *ACM SIGCOMM Computer Communication Review*, vol. 36, pp. 87–92, Jan. 2006.

[6] D. Alderson, L. Li, W. Willinger, and J. C. Doyle, "Understanding Internet topology: principles, models, and validation," *IEEE/ACM Transactions on Networking*, vol. 13, no. 6, pp. 1205–1218, Dec. 2005.

[7] A. V. Aho and J. E. Hopcroft, *The Design and Analysis of Computer Algorithms*. Addison-Wesley Longman Publishing Co., Inc., 1974.

[8] H. Hisamatu, H. Ohsaki, and M. Murata, "Steady state and transient state behaviors analyses of TCP connections considering interactions between TCP connections and network," *International Journal of Communication Systems*, vol. 18, pp. 619 – 637, Sept. 2005.

[9] A. Kowalski and B. Warfield, "Modelling traffic demand between nodes in a telecommunications network," in *Australian Telecommunications and Networks Conference (ATNC)*, Dec. 1995.

[10] H. Hisamatu, H. Ohsaki, and M. Murata, "Fluid-based analysis of a network with DCCP connections and RED routers," in *Proceedings of International Symposium on Applications and the Internet (SAINT 2006)*, Jan. 2006, pp. 22–29.

[11] J. Padhye, V. Firoiu, D. Towsley, and J. Kurose, "Modeling TCP throughput: a simple model and its empirical validation," in *Proceedings of ACM SIGCOMM*, Sept. 1998, pp. 303–314.