PAPER

Performance Analysis of Large-Scale IP Networks considering TCP Traffic

Hiroyuki HISAMATSU[†a)], ***Student Member***, **Go HASEGAWA**[††b)], ***and*** **Masayuki MURATA**[†c)], ***Members***

SUMMARY In this paper, we propose a novel analysis method for large-scale networks with consideration of the behavior of the congestion control mechanism of TCP. In the analysis, we model the behavior of TCP at end-host and network link as an independent system, and combine them into a single system in order to analyze the entire network. Using this analysis, we can analyze a large-scale network, i.e. with over 100/1,000/10,000 routers/hosts/links and 100,000 TCP connections in substantially short time. Especially, a calculation time of our analysis, it is different from that of ns-2, is independent of a network bandwidth and/or propagation delay. Specifically, we can derive the utilization of the network links, the packet loss ratio of the link buffer, the round-trip time (RTT) and the throughput of TCP connections, and the location and degree of the network congestion. We validate our approximate analysis by comparing analytic results with simulation ones. We also show that our analysis method treats the behavior of TCP connection in a large-scale network appropriately.

key words: *TCP (Transmission Control Protocol), Large-Scale Network, Fluid Flow Approximation*

1. Introduction

In recent years, the numbers of internet nodes/hosts and internet users have been increasing exponentially. For example, the number of computers connected to the Internet was about 250 million in February 2004, whereas by January 2005 it had increased to about 350 million. This means that the number of internet hosts has increased by about 40% in only 11 months [1]. Consequently, the importance of design and performance analysis techniques for large-scale networks is increasing. However, currently, there are no effective methods for analyzing such large-scale networks.

One important factor for determining the performance of the Internet is the congestion control mechanisms of TCP. One reason for this is that TCP traffic accounts for a large proportion of current internet traffic [2]. However, when considering the design and performance analysis issues of a large-scale network, the TCP congestion control mechanism, which is based on a feedback control, has been neglected. Most previous studies on large-scale network design assume that the constant-rate UDP flows as traffic demand [3-5]. For example, in [5], the authors revealed that the router-level topology of the current Internet follows a power-law distribution, as a consequence of seeking to maximize the throughput of the network subject to technological constraints on link capacities, packet processing speeds, and the number of input/output links. However, in [5], only the UDP flow with constant bit rate is considered as network traffic, while the packet loss in a network is ignored. That is, it does not include the effect of the behavior of TCP, which uses packet loss as feedback information from the network and regulates the packet transmission rate.

On the other hand, there have been some studies done on the relationship between the congestion control mechanisms of TCP and network performance [6-8]. In these studies, the authors utilized TCP traffic as network traffic and revealed in detail various characteristics on the interaction between TCP behavior and the underlying networks. However, in most of these studies, the number of TCP connections which can be treated is limited to thousands, and very simple network topologies, such as a dumbbell-type network, are used. One of the reasons for this may be the limitations of network simulators such as ns-2 [9].

There have also been many studies on methods for analyzing a large-scale network and many flows modeled using a fluid-flow approximation [10-12]. For example, in [10], a performance evaluation technique for large-scale networks using a fluid approximation model has been proposed. In [10], the congestion control mechanisms of TCP and the active queue management mechanism are modeled. In addition, the effect of the routers' packet processing speed is also modeled through explicit modeling of the order of the routers in which each TCP connection traverses. However, to the best of our knowledge, these studies are currently in the establishment phase in terms of creating analysis methods. As such, there is no means of finding out the interaction effect of a large number of TCP connections, i.e. with over 10,000 connections, and a large-scale network with over 100/1,000/1,000 routers/hosts/links.

In this paper, we propose a novel analysis method for such large-scale networks that takes into consideration the behavior of the congestion control mechanism of TCP. In the analysis, we model each network component (end-host's TCP and network link) as an independent system, and then combine them into one system in order to analyze the entire network. Note that we assume many TCP flows on each network link and utilize appropriate modeling methods based on this assumption. Using this analysis, we can analysis a large-scale network, i.e. with over 100/1,000/10,000

[†] The authors are with Department of Information Networking, Graduate School of Information Science and Technology, Osaka University.

[††] The author is with Department of Cyber Media Center, Osaka University

a) E-mail: hisamatu@ist.osaka-u.ac.jp

b) E-mail: hasegawa@cmc.osaka-u.ac.jp

c) E-mail: murata@ist.osaka-u.ac.jp

routers/hosts/links and 100,000 TCP connections in substantially short time. Especially, a calculation time of our analysis, it is different from that of ns-2, is independent of a network bandwidth and/or propagation delay. Specifically, we derive the utilization of the network link, packet loss ratio of the link buffer, the round-trip time (RTT) and throughput of TCP connections, and the location and the degree of the network congestion. Consequently, we are then in a position to answer the following questions based on the analysis results: When the network traffic increases, which link will become congested? Which access networks and core networks are bottlenecks for the entire network? Which part of the network should be upgraded when we want to increase the network performance? If networking technologies in access/core networks, such as the link bandwidth and the number of input/output ports of routers, are improved, how will the congestion points of the network move (or will they remain unchanged)? Furthermore, will the end-to-end TCP throughput increase as we expect? By answering the above questions, we can use the proposed analysis method to design future high-speed and large scale networks.

This paper is organized as follows. In Section 2, we introduce the network model and traffic models used in this paper. In Section 3, we describe the analysis methods used in modeling a TCP and a network link as independent systems. By combining these models, we obtain the model for the entire network. In section 4, we present several numerical examples of the analysis results. First, we validate our proposed analysis technique by comparing the analytic results with simulation results. Next, we show some analysis results for large scale networks and demonstrate the validity of the proposed analysis method. Finally, in Section 5, we conclude the paper and discuss future work.

2. Network and Traffic Models

In this section, we introduce the models of network and traffic used in this paper. In the analysis, we analyze the average behavior of the entire network when there are many TCP connections present.

2.1 Network Model

Figure 1 shows the network model used in the analysis. The model consists of nodes and links, where the nodes correspond to a host or a router, and the links to links between routers and hosts. 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, link (v, w) differs from 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 propagation delay of link (v, w) are denoted by $\mu_{(v,w)}$ and $\tau_{(v,w)}$, respectively. In this analysis, 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 $b_{(v,w)}$.

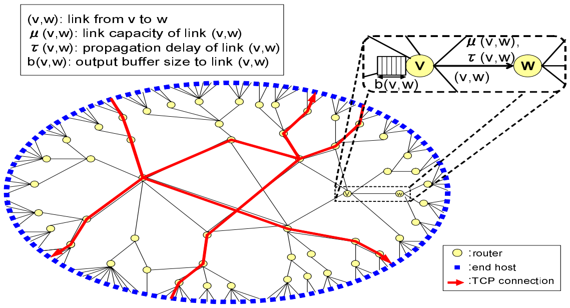

(v,w): link from v to w
μ (v,w): link capacity of link (v,w)
τ (v,w): propagation delay of link (v,w)
b(v,w): output buffer size to link (v,w)
μ (v,w),
τ (v,w)
(v,w)
b(v,w)
v
w
v
w
:router
:end host
:TCP connection

Fig. 1: Network Model in the Analysis

Table 1 Notations for network model

$\mathcal{R}$	set of source and destination hosts and routers
$\mathcal{L}$	set of links
$\mathcal{L}(\chi)$	set of links that TCP connection χ traverses
$\mu_{(v,w)}$	capacity of link (v, w)
$\tau_{(v,w)}$	propagation delay of link
$b_{(v,w)}$	output link buffer size to (v, w) at node v
$\mathcal{C}$	set of TCP connections
$\mathcal{C}(v, w)$	set of TCP connections traversing link (v, w)

TCP connections are established between end hosts according to the amount of traffic defined in Section 2.2. $\mathcal{C}$ is a set of TCP connections. After determining the route which 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 4 we use Dijkstra's shortest path algorithm for determining the route which each TCP connection traverses. Note that we could apply any kind of routing algorithm. For example, we could evaluate the effect of the overlay routing algorithm by applying the algorithm to the TCP connections which join the overlay network. We summarize the notations employed in the network model in Table 1.

In this paper, we use a Drop-Tail discipline at a router buffer, and focus on the average behavior of queue occupancy at the router 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 the appropriate model to the router buffer. For example, for RED discipline, we can use the existing model in [8].

2.2 Traffic Model

The amount of network traffic is determined using the "gravity model" [13]. 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 router to which the end host is connected. In what follows, we call the router an "edge router". We also assume that the amount of traffic injected into/leaving from the edge router is proportional to the number of end hosts connected to the edge router. Finally, the number of TCP connections between edge routers is taken to be proportional to the amount of traffic between the edge routers. The number of TCP connections that traverse from edge router v to w is defined as

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

where E_v and E_w are the numbers of end hosts connected to the edge routers v and w, respectively, and α is a 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. Moreover, we can also analyze a network which has different TCP versions in the same network. Hereafter, TCP Reno is simply denoted as TCP unless noted otherwise.

3. Analysis

In the analysis, we first model a TCP and a network link as independent systems. 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 for decreasing the complexity of the simultaneous equations by removing links which do not cause congestion.

3.1 Modeling of TCP Behavior

We focus on the average behavior of a TCP connection, which varies the average window size depending on the packet loss ratio. That is, we model a TCP connection as a system with one input (packet loss ratio) and one output (average window size). Given a packet loss ratio d_χ and a RTT r_χ of a TCP connection χ, λ_χ, the average throughput of a TCP connection, can be calculated using the following result [14];

$$\lambda_\chi = \frac{1}{r_\chi\left(\sqrt{\frac{2d_\chi}{3}} + 6\sqrt{\frac{3b\,d_\chi}{2}}d_\chi(1+32d_\chi^2)\right)}, \tag{2}$$

where b is the number of required data packets for a TCP receiver to generate one ACK packet, and T_o is the initial value of the TCP retransmission timeout. By applying $b = 1$ and $T_o = 4\,r_\chi$ [15], the average size of the congestion window of TCP connection χ, denoted by w_χ, can be given by;

$$w_\chi = \frac{1}{\sqrt{\frac{2p_\chi}{3}} + 6\sqrt{\frac{3b\,p_\chi}{2}}p_\chi(1+32p_\chi^2)}. \tag{3}$$

Table 2 Notations for TCP model

w_χ	congestion window size of TCP connection χ
τ_χ	round trip propagation delay of TCP connection χ
r_χ	RTT of TCP connection χ
d_χ	packet loss ratio of TCP connection χ
λ_χ	throughput of TCP connection χ
$d_{(v,w)}$	packet loss ratio at output buffer of link (v,w)
$q_{(v,w)}$	number of packets in output link buffer of link (v,w)

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

$$d_\chi = 1 - \prod_{(v,w)\in\mathcal{L}(\chi)} (1 - d_{(v,w)}), \tag{4}$$

We can also derive r_χ and τ_χ, which are the RTT of the TCP connection χ and the round-trip propagation delay of the TCP connection r_χ which does not include the queuing delay at traversing links, respectively, as follows;

$$r_\chi = \tau_\chi + \sum_{(v,w)\in\mathcal{L}(\chi)} \frac{q_{(v,w)}}{\mu_{(v,w)}} \tag{5}$$

$$\tau_\chi = \sum_{(v,w)\in\mathcal{L}(\chi)} \tau_{(v,w)} \tag{6}$$

We summarize the notations used in this subsection in Table 2.

3.2 Modeling of Network Link

We focus on the behavior of a network link when TCP connections, which have certain values 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 one output (packet loss ratio).

In [16], the authors have revealed the following characteristic on 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 having a large number of TCP connections, we utilize the above characteristics. Then, we can calculate $d_{(v,w)}$, the packet loss ratio at the buffer of link (v,w), as follows;

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

where $\sigma(q_{(v,w)})$ is the standard deviation of the distribution of the number of packets in the output link buffer of link (v,w), and $Erf()$ is the error function. The analysis in [16] assumes that the standard deviation of the distribution of the

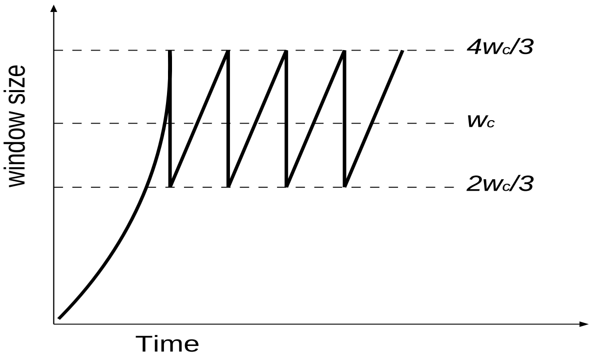


Fig. 2: Evolution of TCP congestion window

number of packets in the output link buffer is identical to that of the sum of the congestion window size of the TCP connections traversing the link. We therefore derive $d_{(v,w)}$ based on this assumption as follows.

Figure 2 depicts the typical change in the congestion window size of a TCP connection. 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 $2w_\chi/3$ and an upper limit of $4w_\chi/3$, where w_χ 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;

$$\sigma(w_\chi) = \frac{w_\chi}{3\sqrt{3}}.$$

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

$$\sigma\left(\sum_{\chi\in\mathcal{C}(l_{v,w})} w_\chi\right) = \sigma\left(\sqrt{\sum_{\chi\in\mathcal{C}(l_{v,w})} \sigma(w_\chi)^2}\right). \quad (8)$$

In addition, we utilize the assumption that when link (v, w) is congested, the sum of the throughput of TCP connections traversing link (v, w) becomes the link capacity $\mu_{(v,w)}$;

$$\mu_{(v,w)} = \sum_{\chi\in\mathcal{C}(v,w)} \lambda_\chi. \quad (9)$$

3.3 Connecting Systems and Analysis

We regard Equations (2) – (6) and ((8) – (9)) as simultaneous equations, and solve them for w_χ, $d_{(v,w)}$, and $q_{(v,w)}$. We then obtain the window size and throughput of each TCP connection, the number of packets in the output link buffer and the packet loss ratio at each network link. The straightforward nature of the analysis is one of the advantages of our analysis method.

3.4 Reduction of Analysis Model

In the actual network, the number of congested links is not as large as the total number of links. The number of packets in the buffer and the packet loss ratio of uncongested links become zero. By removing such uncongested links from the analysis calculation, we can reduce the calculation time. In our analysis, we utilize the following method for reducing the number of links from the analysis. Note that the following method is based on the similar method in [10], but we have extended the method to accommodate TCP traffic.

1. Calculate the "maximum" throughput of each TCP connection.

   A maximum throughput $max\lambda_\chi$ of TCP connection χ traverses link (v, w), where v is the source node of TCP connection χ and is defined as follows;

   $$max\lambda_\chi = \mu_{(v,w)} \times \frac{\frac{1}{\tau_\chi}}{\sum_{\psi\in\mathcal{C}(v,w)} \frac{1}{\tau_\psi}}.$$

2. Calculate the "maximum" amount of traffic of each network link.

   We take the maximum amount of traffic $maxT_{(v,w)}$ of link (v, w) to be the sum of the maximum throughput $\lambda_\psi(\psi \in \mathcal{C}(v, w))$ of TCP connections traversing the link,

   $$maxT_{(v,w)} = \sum_{\psi\in\mathcal{C}(v,w)} max\lambda_\psi$$

3. Compare the maximum amount of traffic with the bandwidth at each link.

   case 1 $\forall(v, w) maxT_{(v,w)} \le \mu_{(v,w)}$.
   There is no congestion at links which satisfy $maxT_{(v,w)} < \mu_{(v,w)}$. We remove these uncongested links from the analysis model and reduce the complexity of the model.

   case 2 $\exists(v, w) maxT_{(v,w)} > \mu_{(v,w)}$.
   We change the maximum throughput of the TCP connections. We focus on link (v, w), which has the maximum difference between the maximum amount of traffic $maxT_{(v,w)}$ and the bandwidth $\mu_{(v,w)}$ of link (v, w). Then, we change the maximum throughput of TCP connections as follows. Let $\alpha_\chi(\chi \in \mathcal{C}(v, w))$ be the maximum throughput of TCP connections. We divide the link capacity $\mu_{(v,w)}$ by the inverse ratio of propagation delays of TCP connections traversing (v, w). We denote them as $\beta_\chi(\chi \in \mathcal{C}(v, w))$. In the case of $\alpha_\chi \le \beta_\chi$, we assign β_χ to α_χ. Then, we repeat Step 2.

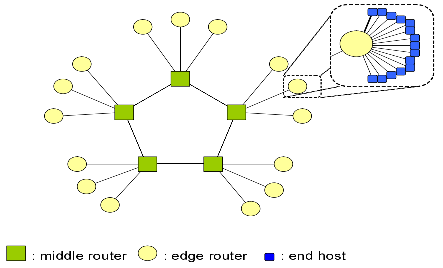
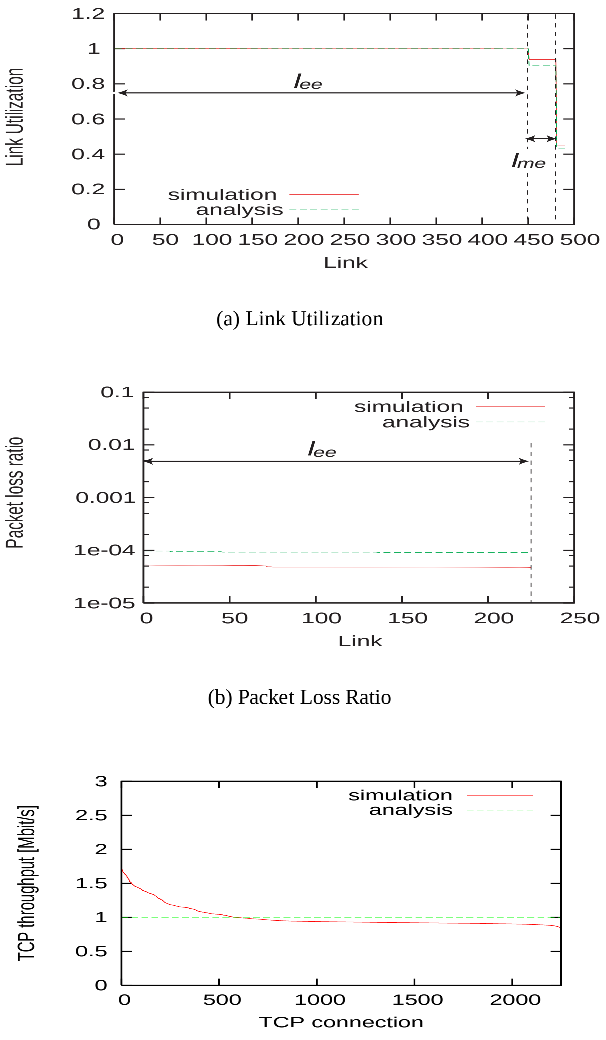


Fig. 3: Network model for testing the accuracy of the analysis method

Table 3 Parameter Settings (1)

Link	Bandwidth	Prop. Delay	Buffer Size
l_{mm}	622 [Mbit/s] (OC12)	5 [ms]	1,043 [packet]
l_{me}	155 [Mbit/s] (OC3)	5 [ms]	850 [packet]
l_{ee}	10 [Mbit/s]	10 [ms]	1,500 [packet]



(a) Link Utilization

(b) Packet Loss Ratio

(c) TCP Throughput

Fig. 4: Access link bandwidth = 10 [Mbit/s]

4. Numerical Examples

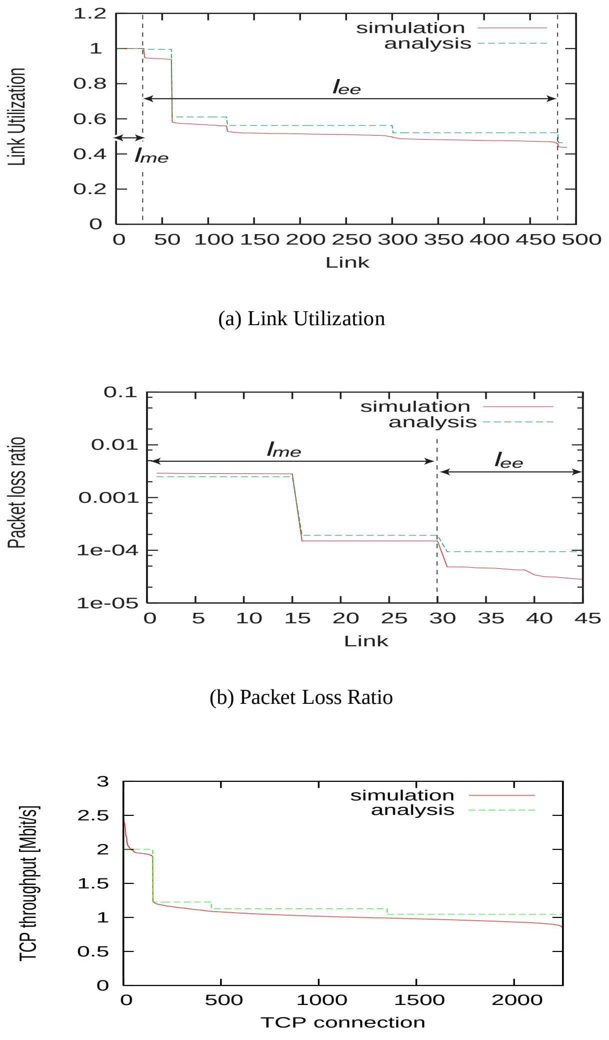
In this section, we verify the accuracy of the analysis method by comparing the analysis and simulation results. We then show analytic results for large-scale networks and demonstrate the ability of the proposed analysis method.

4.1 Accuracy of Analysis Method

We use the network model depicted in Fig. 3 for assessing the accuracy of the analysis method. The network topology consists of "middle routers", "edge routers" and "end hosts". For simplicity, we denote links between middle routers as l_{mm}, those between middle routers and edge routers as l_{me}, and those between edge routers and end hosts as l_{ee}.The bandwidth, propagation delay, and output link buffer size were set to the values shown in Tab. 3. We set α in Eq. 1 to $2/45$. In this setting, the total number of TCP connections in the network becomes $2,250$. The number of TCP connections between edge routers is determined by the gravity-model introduced in Section 2.2. To obtain the simulation results, we utilized an ns-2 simulator and conducted the simulation using the same network model as of the analysis. The simulation time was 1,050 [s], and we omit the results for the initial 50 [s] to avoid the effect of unstable behavior at the beginning of the simulation. The packet size was set to 1,000 [bytes]. Our experiment is carried on a Dell powerEdge 1850, which has two Intel Xeon processors (3.80GHz) and 4GB memory. Note that we can not conduct the ns-2 simulation for the larger scale networks than in Fig. 3.

Figures 4 and 5 show the analytic and simulation results when the bandwidth of the access links is set to 10 and 20 [Mbit/s], respectively. Figures 4(a) and 5(a) plot the utilization of the links. Figures 4(b) and 5(b) show the packet loss ratio of links having non-zero packet loss ratio and Figs. 4(c) and 5(c) show the throughput of TCP connections in the network. In the simulation results of these figures, we plot the link utilization and the packet loss ratio and the TCP throughput in decreasing order of their value. In the analytic results of these figures, we plot them in the same order as those of the the analytic results to compare the analytic result with the simulation ones. We also add which type of the link gives the corresponding results in Figs. 4(a),(b), and 5(a),(b).

It can be found that the analytic results results give the close estimation of the simulation results. In particular, in respect to the utilization of the links, it can be found that our analytic results show very good agreement with the simulation results. However, in terms of the TCP throughput,



(a) Link Utilization

(b) Packet Loss Ratio

(c) TCP Throughput

Fig. 5: Access link bandwidth = 20 [Mbit/s]

it can be found that our analytic results are different from simulation results especially when the TCP throughput of the simulation results is comparatively large (Figure 4(c)). This is because of the deviation of packet loss occurrences among TCP connections in simulation. That is, packets of “lucky TCP connections” are seldom lost, and many packets of “unlucky TCP connections” are lost, which brings the fluctuation of the throughput of TCP connections. In other words, the 1,050 [s] of the simulation time is small to obtain the average throughput of TCP connections in this situation. This is one of the shortcomings of the simulation, whereas the analysis method in this paper directly gives the result of the average TCP throughput.

From Fig. 4, we can see that when the bandwidth of the l_{ee} is 10 [Mbit/s], the bottleneck point in the network is the link l_{ee}. Note that our analytic results and simulation ones show that the same links are bottlenecks. From Figs. 5, we can see that when the bandwidth of the l_{ee} is 20 [Mbit/s], the bottleneck point in the network moves to the link l_{me}, Note that our analytic and simulation results find the same links as bottleneck links. From the above results, we conclude that our analysis can precisely determine the bottleneck point precisely.

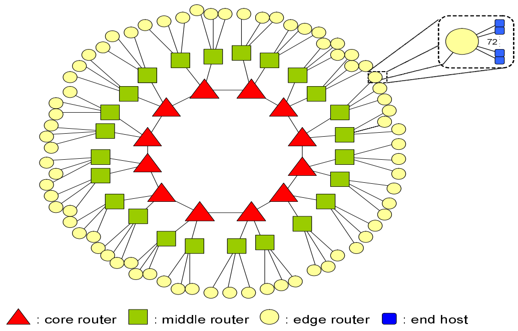


Fig. 6: Network model for analysis of large-scale network

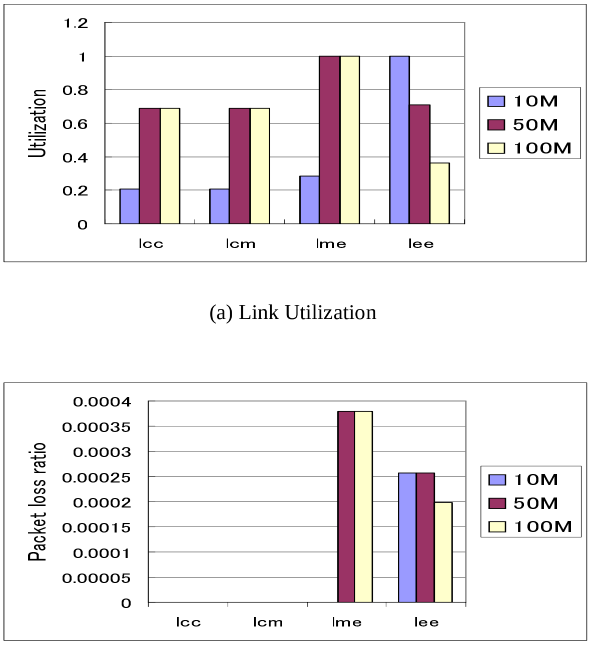
4.2 Analysis Results of Large-Scale Network

In this subsection we give examples of the analytic results on the large-scale network. Figure 6 shows the network model used in the analysis. This network topology was created according to the characteristics of the actual router-level topology, which was described in [5], 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 network topology consists of “core routers”, “middle routers”, “edge routers” and “end hosts”. 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 end hosts as l_{ee}. The bandwidth, propagation delay, and output link buffer size were set to the values shown in Tab. 4. Note that the buffer sizes of all links except l_{ee} were set according to the guidelines given in [16], where the number of connections at l_{cc}, l_{cm}, and l_{me} was assumed to be 10,000, 1,000, and 1,000, respectively, and the average RTT of TCP connection was assumed to 150 (ms). We set α in Eq. 1 to $20/5,184$. In this setting, the total number of TCP connections in the network becomes $103,680$. The number of TCP connections between edge routers is determined by the gravity-model as in the previous subsection. It is unable for the ns-2 simulator to carry out the simulation of this scale of network.

Figure 7 shows the analytic results when the bandwidth of l_{ee}, which is the access link bandwidth, had the values 10, 50, and 100 [Mbit/s]. Figures 7(a) and (b) plot the average link utilization and the average packet loss ratio of the links l_{cc}, l_{cm}, l_{me} and l_{ee}, respectively. From Fig. 7(a), we can observe that the utilization of the links inside the net-

Table 4 Parameter Settings (2)

Link	Bandwidth	Prop. Delay	Buffer Size
l_{cc}	10 [Gbit/s] (OC192)	15 [ms]	3,750 [packet]
l_{cm}	2.5 [Gbit/s] (OC48)	5 [ms]	2,370 [packet]
l_{me}	1 [Gbit/s] (GE)	5 [ms]	1,185 [packet]
l_{ee}	10 [Mbit/s]	10 [ms]	1,500 [packet]



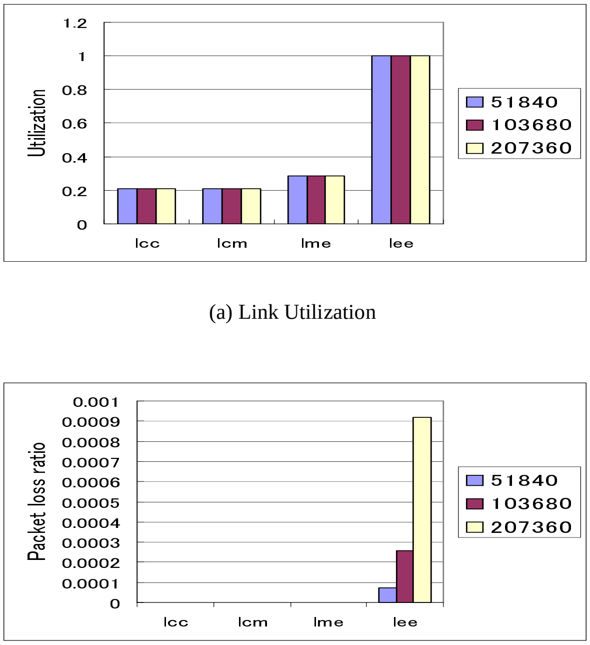
(a) Link Utilization

(b) Packet Loss Ratio

Fig. 7: Effect of access link bandwidth

work (l_{cc}, l_{cm}, and l_{me}) increases as the bandwidth of l_{ee} increases. Furthermore, we can see that when the bandwidth of the link l_{ee} is 10 [Mbit/s], the bottleneck point in the network is the link l_{ee}, but when it is increased to 50 [Mbit/s], the bottleneck point moves to the link l_{me}. This movement of the bottleneck point is confirmed by Fig. 7(b), where the packet loss probability of l_{me} is more than that of l_{ee} when the l_{ee} is 50 [Mbit/s]. These results mean that by increasing the access link bandwidth, the capacity of the core network becomes comparatively small. We can see such a situation easily by using the analytic results, without the need for time-consuming simulation experiments.

We next show the results when we vary the number of TCP connections in the network to 51,840, 103,680, and 207,360, by changing α to $10/5, 184$, $20/5, 184$, and $40/5, 184$, respectively. We set the bandwidth of the link l_{ee} to 10 [Mbit/s]. Figures 8(a) and 8(b) show the average link utilization and the average packet loss ratio of the links l_{cc}, l_{cm}, l_{me}, and l_{ee}. We can determine the following network characteristics of the networks from the analytic results. The link utilization remains unchanged when the number of TCP connections changes. This clearly shows the greedy nature of the congestion control mechanism of TCP: TCP always tries to fully utilize the link bandwidth when the receive socket buffer size is large enough. The effect of increasing the number of TCP connections is found on the packet loss ratio, shown in Fig. 8(b). This again demonstrates the nature of TCP connections. From these results, we have confirmed that our analysis method can describe the behavior of TCP connections in a large-scale network appropriately.



(a) Link Utilization

(b) Packet Loss Ratio

Fig. 8: Effect of the number of TCP connections

5. Conclusion and Future Work

In this paper, we have proposed a novel analysis method for such large-scale networks with consideration of the behavior of the congestion control mechanism of TCP. In the analysis, we have modeled each network component (end-host's TCP and network link) as a independent system, and interconnect them into one system for analyzing the entire network. By the analysis, we have derived the utilization of the network link, packet loss ratio of the link buffer, the round-trip time and throughput of TCP connections, and the location and the degree of the network congestion. By showing some numerical examples, we have shown that our analysis method can treat the behavior of TCP connection in the large-scale network appropriately.

In recent years, data transmission between the end-hosts may be carried out via overlay nodes by dividing the end-to-end TCP connection into multiple split TCP connections. It would be capable to apply our proposed analysis method to design a overlay network. For future work, we plant to resolve the "location of overlay nodes problem" and "path between overlay nodes choice problem" by the analysis method in this paper. We use Dijkstra's shortest path algorithm for determining the route which each TCP connection traverses. It would be interesting to use other routing algorithms and show how the end-to-end TCP throughput changes. For instance, evaluating TCP throughput when using ETR (Estimated-TCP-throughput Maximization based Routing) algorithm [17] for determining the route in a large-scale network would be interesting. It would be important to analyze a network which has new TCP variants proposed for high-speed and large-delay networks. By using our proposed analysis method, we can investigate the influence of such TCP variants to a large-scale network; how will the congestion points of the network move by introducing such TCP variants ? Our analysis can be easily applied to such a situation. In case of HSTCP [18], for example, we can easily model the throughput of an HSTCP by extending the approach proposed in [14].

References

[1] "Hobbes' Internet timeline v8.1." available at http://www.zakon.org/robert/internet/timeline/.

[2] M. Fomenkov, K. Keys, D. Moore, and k claffy, "Longitudinal study of Internet traffic from 1998–2003," Winter International Symposium on Information and Communication Technologies (WISICT 2004), Jan. 2004.

[3] Y. Zhang, M. Roughan, N. Duffield, and A. Greenberg, "Fast accurate computation of large-scale IP traffic matrices from link loads," Proceedings of the ACM SIGMETRICS, pp.206–217, June 2003.

[4] M. Roughan, M. Thorup, and Y. Zhang, "Traffic engineering with estimated traffic matrices," Proceedings of the 3rd ACM SIGCOMM Conference on Internet Measurement, pp.248–258, Oct. 2003.

[5] L. Li, D. Alderson, W. Willinger, and J. Doyle, "First-principles approach to understanding the Internet's router-level topology," Proceedings of ACM SIGCOMM '04, pp.3–14, Sept. 2004.

[6] S.H. Low, F. Paganini, J. Wang, S. Adlakha, and J.C. Doyle, "Dynamics of TCP/RED and a scalable control," Proceedings of IEEE INFOCOM, June 2002.

[7] S.H. Low, "A duality model of TCP and queue management algorithms," IEEE/ACM Transactions on Networking, vol.11, no.4, pp.525–536, 2003.

[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] "The network simulator – ns2." available at http://www.isi.edu/nsnam/ns/.

[10] Y. Liu, F.L. Presti, V. Misra, D. Towsley, and Y. Gu, "Fluid models and solutions for large-scale IP networks," Proceedings of ACM/SIGMETRICS 2003, pp.91–101, June 2003.

[11] H. Ohsaki, J. Ujiie, and M. Imase, "On scalable modeling of TCP congestion control mechanism for large-scale IP networks," Proceedings of IEEE SAINT 2005, pp.361–369, Feb. 2005.

[12] M.A. Marsan, M. Garetto, P. Giaccone, E. Leonardi, E. Schiattarella, and A. Tarello, "Using partial differential equations to model TCP mice and elephants in large IP networks," IEEE/ACM Transactions on Networking, vol.13, no.6, dec 2005.

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

[14] J. Padhye, V. Firoiu, D. Towsley, and J. Kurose, "Modeling TCP throughput: a simple model and its empirical validation," Proceedings of ACM SIGCOMM '98, pp.303–314, Sept. 1998.

[15] M. Handly, S. Floyd, J. Padhye, and J. Widmer, "TCP friendly rate control (TFRC): protocol specification," Request for Comments (RFC) 3448, Jan. 2003.

[16] G. Appenzeller, I. Keslassy, and N. McKeown, "Sizing router buffers," Proceedings of ACM SIGCOMM '04, pp.281–292, Sept. 2004.

[17] H. Takahashi, M. Saito, H. Aida, Y. Tobe, and H. Tokuda, "Estimated-tcp-throughput maximization based routing," Proceedings of IEEE International Conference onLocal Computer Networks (LCN), pp.120–129, 10 2003.

[18] S. Floyd, "Highspeed TCP for large congestion windows," Request for Comments (RFC) 3649, Dec. 2003.

Hiroyuki HISAMATSU received the M. E. degree in the Information and Computer Sciences from Osaka University, Osaka, Japan, in 2003. He is currently a graduate student at the Graduate School of Information Science and Technology, Osaka University. His research work is in the area of performance evaluation of TCP/IP networks. He is a student member of IEEE and IEICE.

Go HASEGAWA received the M.E. and D.E. degrees in Information and Computer Sciences from Osaka University, Osaka, Japan, in 1997 and 2000, respectively. From July 1997 to June 2000, he was a Research Assistant of Graduate School of Economics, Osaka University. He is now an Associate Professor of Cybermedia Center, Osaka University. His research work is in the area of transport architecture for future high-speed networks. He is a member of the IEEE and IEICE.

Masayuki MURATA received the M.E. and D.E. degrees in Information and Computer Sciences from Osaka University, Japan, in 1984 and 1988, respectively. In April 1984, he joined Tokyo Research Laboratory, IBM Japan, as a Researcher. From September 1987 to January 1989, he was an Assistant Professor with Computation Center, Osaka University. In February 1989, he moved to the Department of Information and Computer Sciences, Faculty of Engineering Science, Osaka University. From 1992 to 1999, he was an Associate Professor in the Graduate School of Engineering Science, Osaka University, and from April 1999, he has been a Professor of Osaka University. He moved to Advanced Networked Environment Division, Cybermedia Center, Osaka University in 2000, and moved to Graduate School of Information Science and Technology, Osaka University in April 2004. He has more than two hundred papers of international and domestic journals and conferences. His research interests include computer communication networks, performance modeling and evaluation. He is a member of IEEE, ACM, The Internet Society, IEICE and IPSJ.