

Inconsistency of Logical and Physical Topologies for Overlay Networks and Its Effect on File Transfer Delay

Yasuo Tamura^a, Shoji Kasahara^{a,*}, Yutaka Takahashi^a,
Satoshi Kamei^b, and Ryoichi Kawahara^b

^a*Graduate School of Informatics, Kyoto University
Yoshida-honmachi, Sakyo-ku, Kyoto 606-8501, Japan*

^b*NTT Service Integration Laboratories
Midoricho 3-9-11, Musashino-shi, Tokyo 180-8585, Japan*

Abstract

In overlay networks consisting of underlying physical nodes and links, data files are transferred by overlay routing managed at application level. The end-to-end file-transfer delay depends on both logical and physical topologies resulting from overlay routing. In this paper, we focus on six types of four-terminal four-router physical network, and investigate the impact of topology inconsistency on the file-transfer delay. The analysis model is based on a two-layer queueing network, taking into account both logical and physical topologies. The model is validated in comparison with ns-2 simulation. Numerical examples show that the end-to-end file-transfer delay is small when both logical and physical topologies are the same. It is also observed that the end-to-end delay significantly depends on the traffic intensity for some logical topologies, regardless of physical topology.

Key words: Overlay network, topology inconsistency, file-transfer delay, two-layer queueing network

1 Introduction

Overlay networking is considered as a promising paradigm to guarantee quality of service (QoS) over the Internet. Recently, much effort has been devoted to

* Corresponding author. Tel: +81-75-753-5493; Fax: +81-75-753-3358.
Email address: kasahara@ieee.org (Shoji Kasahara).

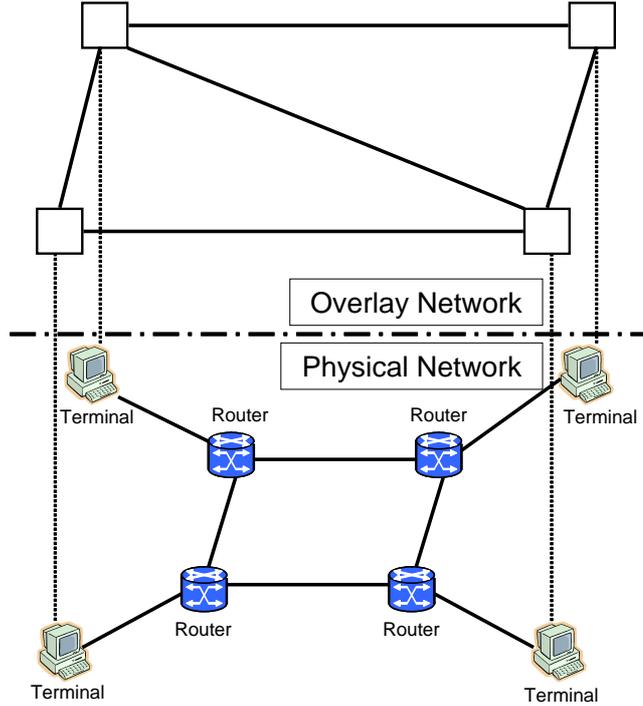


Fig. 1. Four-terminal four-router network.

develop overlay service networks such as resilient overlay network (RON) [1], service overlay network (SON) [2], and QoS-aware routing for overlay networks (QRON) [7]. In general, an overlay network consists of underlying physical nodes such as end hosts, routers and switches. Overlay nodes are connected with overlay-level paths, each of which consisting of one or more physical links.

In the overlay network, data files are transferred by overlay routing managed at application level. Because the overlay network is not aware of its underlying physical network, the logical topology managed at overlay level is not always the same as the underlying physical network topology. It has been reported in [8] that the end-to-end file-transfer delay significantly depends on both logical and physical topologies, and that an inconsistency between logical and physical topologies may cause the degradation of network performance.

In this paper, we consider an analytic framework for evaluating the effect of topology inconsistency on the end-to-end file-transfer delay. To this end, we model the overlay network as a two-layer queueing network [5,6]. A two-layer queueing network is composed of two parts: an upper layer and a lower one. In the two-layer queueing network, a customer arrives at one of upper-layer nodes called stations, and each customer moves from station to station according to an upper-layer routing chain. In each station, the service of a customer is associated with lower-layer nodes, that is, a customer arriving at a station traverses nodes belonging in the lower layer according to a lower-layer routing chain.

Table 1
Topology Classification based on node degree.

Type	Node degree		
	1	2	3
A	0	0	4
B	0	2	2
C	0	4	0
D	1	2	1
E	2	2	0
F	3	0	1

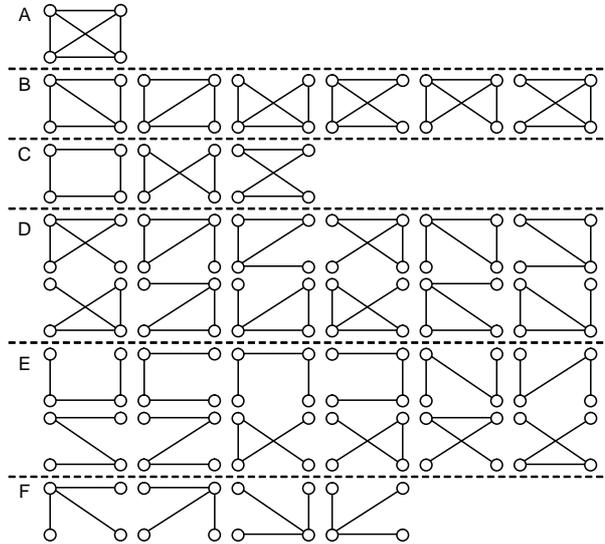


Fig. 2. All topologies of four-node network

In this two-layer queueing network model, however, the computation of performance measures is impracticable when the number of logical and physical nodes is large. Therefore, we focus on four-terminal four-router network as shown in Fig. 1. In this four-terminal four-router network, a physical network consists of four terminals and four routers, while an overlay network is composed of the four terminals.

The number of possible topologies for the four-terminal four-router network is 38. In terms of node degree, which is defined as the number of neighbor nodes, these topologies can be classified into six types. Table 1 shows the topology classification based on node degree. Each value in this table represents the number of nodes that has the node degree specified by column number. For example, type A topology consists of four nodes whose node degree is three. Note that each row sum is equal to four because of four-node network. We also show all possible topologies with respect to each of the topology types

in Fig. 2. Note that types A, C, E and F topologies are conventionally called full-mesh, ring, line and star, respectively. In the following, type B is called five-link, and type D triangle with a branch (TWB).

In this paper, we analyze the end-to-end file-transfer delay for overlay networks by two-layer queueing network models. Six topologies are considered for each layer, and hence 36 combinations of overlay and physical topologies are comprehensively studied. We compare analysis results with those by ns-2 simulation, discussing the efficiency of the analysis based on the two-layer queueing system. Then, we numerically investigate how the end-to-end file transfer delay is affected by the topology inconsistency.

The four-terminal four-router network is not large enough to characterize the end-to-end file-transfer delay observed in real networks such as the Internet. However, the four-terminal four-router network serves several topologies in terms of overlay networking. In this paper, we consider six type topologies for both logical and physical networks, resulting in 36 topology combinations. We investigate the end-to-end file-transfer delay for each of the combinations. This thorough investigation gives us a significant insight into the effect of topology inconsistency on the end-to-end file-transfer delay for overlay networks.

The rest of the paper is organized as follows. In Section 2, we summarize the related work for analytical approach to the performance evaluation of overlay networks. Section 3 describes our analysis model based on two-layer queueing system in detail, and the analysis is presented in Section 4. In Section 5, we present some numerical examples, and finally, Section 6 presents the concluding remark.

2 Related Work

Recently, overlay networking has received considerable attention, and there is much literature on performance issue of overlay networks including peer-to-peer (P2P) networks and content distribution networks (CDN's).

In [8], the effect of overlay topology on the performance of overlay routing service was studied by simulation. Here, the authors investigated routing performance in terms of failure recovery and routing overhead. It was observed that the overlay topology greatly affects the overlay routing performance and that the awareness of underlying physical network topology improves the performance significantly.

In [9], the interaction between the overlay-level routing and the routing based on traffic engineering (TE) was studied by a game theoretical approach. It

was reported that the objective misalignment between overlay and TE causes oscillations in their routes, which significantly degrade end-to-end network performance.

In terms of mathematical modeling, much efforts have been devoted to characterize application-level performance over P2P networks. Ge et al. [3] studied the system throughput and response time for P2P file sharing service. The authors considered P2P file sharing service as a closed queueing network, and analyzed the performance of P2P file sharing service for three fundamental architectures: centralized indexing, flooding-based indexing, and distributed indexing with hashing directed queries.

The work in [12] and [10] analyzed the scaling effect of P2P file sharing service based on BitTorrent. In [12], the service capacity for BitTorrent-like P2P file sharing networks was studied. The transient state was analyzed with a branching process, while the steady state was investigated by a simple Markovian model. In [10], the scaling effect of BitTorrent-like P2P file sharing service was studied by a deterministic fluid model, and the incentive mechanism of BitTorrent was analyzed through a game theoretical approach.

In [11], the authors assumed that dominant factors of the file transfer delay for a P2P file sharing network are network-level latency and peer-level one. The former was modeled as a single class open queueing network consisting of routers, while the latter was modeled as an M/G/1/K processor sharing queue. This was the first work taking into account the effect of physical network topology on the end-to-end file transfer delay. Note that in P2P file sharing service, a file is transferred with a single-hop connection at overlay network level. In our study, we focus on the overlay routing in which an end-to-end logical path may consist of several overlay nodes, and hence the file-transfer delay is significantly affected by the network topology at overlay level.

The authors in [13] considered the ability of an overlay network to compensate for careless routing in underlay network level. The routing performance of the underlay network was characterized by the characteristic path length, the cut average and the weighted sum of node degrees. It was found that the effectiveness of an overlay compensating for the careless underlay depends on both the overlay and the underlay. Note that this work considered the routing performance issue from a graph theoretical point of view. In our study, we consider the delay performance of the topology mapping which results from both overlay and underlay routing optimizers, assuming the shortest hop-count routing on both overlay and underlay layers.

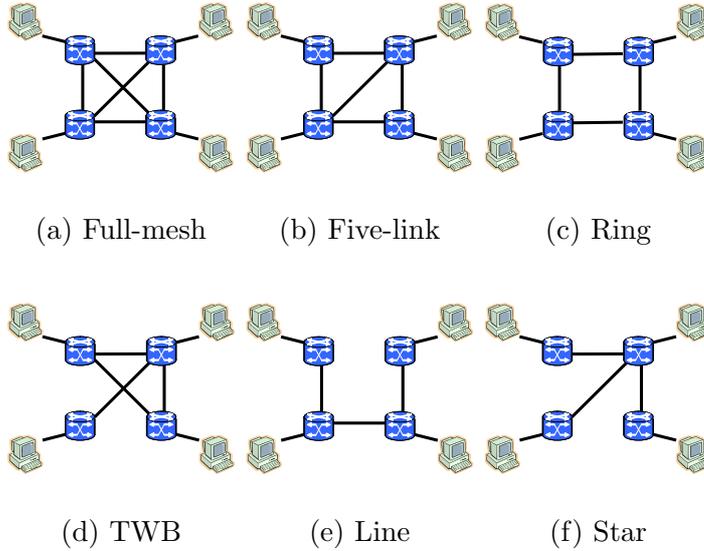


Fig. 3. Six types of four-terminal four-router physical network.

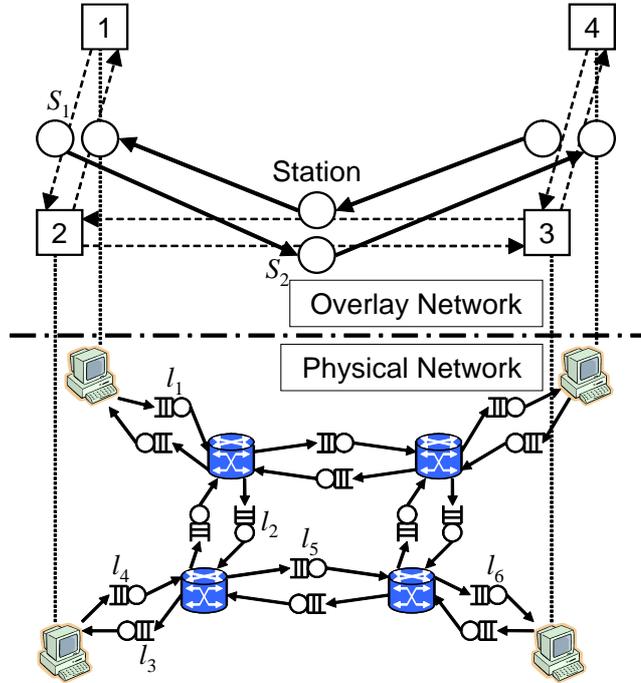


Fig. 4. Two-layer queueing network model for four-terminal four-router physical network.

3 Analysis model

3.1 Two-layer queueing network model

In this paper, we focus on the following six types of network topology: full-mesh, five-link, ring, TWB, star, and line topologies. (See Fig. 3.) We char-

acterize end-to-end file transmission in an overlay network with a two-layer queueing network. Remind that a physical network is composed of four terminals and four routers, while an overlay network consists of four terminals. The key idea for modeling overlay networks is that a logical path established between two overlay nodes is modeled as a station.

Figure 4 shows an example of a two-layer queueing network. In this figure, a rectangle with number in the logical network denotes an overlay node, which is corresponding to a terminal node in the physical network. A dashed arrow in the logical network represents a logical path between two overlay nodes. In our two-layer queueing network model, each logical path is modeled as a station (a circle in the logical network in the figure).

Each logical path in Fig. 4 is composed of the physical links along with the logical path corresponding to the station. When a file is transferred from one overlay node to the other, the corresponding file transfer time consists of the sum of transmission times of the physical links along with the logical path. We model each physical link as a queueing system.

In the logical network, an end-to-end file transmission is managed with a routing chain at overlay level. At each station in the logical network, a file traverses through queues in the physical layer according to a routing chain depending on the station the file visits.

Now consider the case where a file is transferred from overlay node 1 to node 3 in Fig. 4. The file is transferred from overlay node 1 to node 2 first, and then moved from node 2 to node 3. In this case, the corresponding route in the upper layer is station S_1 to station S_2 . Note that this route is specified by the routing chain at overlay level. In terms of the physical network, S_1 consists of physical links l_1 , l_2 and l_3 , while S_2 is composed of l_4 , l_5 , and l_6 . The file traverses through l_1 , l_2 and l_3 , and reaches node 2. Then it traverses through l_4 , l_5 , and l_6 , and finally arrives at node 3.

3.2 Model description

This subsection gives model description and notations in detail.

Suppose that there are N_M terminals and hence N_M overlay nodes in the network system. In our two-layer queueing model, there are N_I stations in the upper layer and N_J queues in the lower layer. The outside of the network is labeled 0. We assume that each station in the upper layer is an infinite server queue.

A file request arrives at overlay node m ($1 \leq m \leq N_M$) according to a Poisson

process with rate λ_m . We call a file whose request arrives at overlay node m a type- m file. Let $s_{i_1 i_2}(m)$ ($0 \leq i_1, i_2 \leq N_I$) denote the probability that a type- m file at station i_1 moves to station i_2 . Note that a type- m file arriving from outside of the network joins station i_2 with probability $s_{0i_2}(m)$, and that a type- m file at station i_1 leaves the network with probability $s_{i_1 0}(m)$. We define the Markovian routing chain of type- m customers in the logical network as

$$\mathbf{S}(m) = \begin{pmatrix} 0 & s_{01}(m) & \dots & s_{0N_I}(m) \\ s_{10}(m) & s_{11}(m) & \dots & s_{1N_I}(m) \\ \vdots & \vdots & \ddots & \vdots \\ s_{N_I 0}(m) & s_{N_I 1}(m) & \dots & s_{N_I N_I}(m) \end{pmatrix}. \quad (1)$$

Let G_m ($1 \leq m \leq N_M$) be a set of stations a type- m file may visit. Note that

$$s_{i0}(m) = 1 - \sum_{j \in G_m} s_{ij}(m), \quad \sum_{j \in G_m} s_{0j}(m) = 1, \quad m = 1, 2, \dots, N_M.$$

When a type- m file visits station i , the type- m file traverses through queues in the physical network, which are associated with station i . We call the file in the physical network the station- i file, regardless of its type.

The capacity of queue j ($1 \leq j \leq N_J$) in the physical network is infinite and its transmission rate is μ_j . We assume that the service discipline of each queue is processor sharing in order to characterize bandwidth share due to Transmission Control Protocol (TCP). That is, when there are n files in transmission at queue j , the transmission rate of each file is given by μ_j/n .

We define $r_{j_1 j_2}(i)$ ($0 \leq j_1, j_2 \leq N_J$, $1 \leq i \leq N_I$) as the probability that a station- i file at queue j_1 joins queue j_2 . Here, a station- i file just arriving at station i in the logical layer moves to queue j_2 with probability $r_{0j_2}(i)$, while a station- i file at queue j_1 leaves the physical layer and returns station i with probability $r_{j_1 0}(i)$. Note that the station- i file's return to station i is equivalent to the type- m file's service completion at station i . The next station to be visited by the type- m file is determined by $\mathbf{S}(m)$.

We define the Markovian routing chain of station- i files in the physical network as

$$\mathbf{R}(i) = \begin{pmatrix} 0 & r_{01}(i) & \dots & r_{0N_J}(i) \\ r_{10}(i) & r_{11}(i) & \dots & r_{1N_J}(i) \\ \vdots & \vdots & \ddots & \vdots \\ r_{N_J 0}(i) & r_{N_J 1}(i) & \dots & r_{N_J N_J}(i) \end{pmatrix}. \quad (2)$$

Let H_i ($1 \leq i \leq N_I$) be a set of queues a station- i file may visit. Note also that

$$r_{j0}(i) = 1 - \sum_{k \in H_i} r_{jk}(i), \quad \sum_{k \in H_i} r_{0k}(i) = 1, \quad i = 1, 2, \dots, N_I.$$

We show an example of routing chains $\mathbf{S}(m)$ and $\mathbf{R}(i)$ in Appendix.

In the following, we introduce the following notations for simplicity

$$\begin{aligned} \|\mathbf{a}\| &= a_1 + a_2 + \dots + a_n, \\ \mathbf{a}! &= a_1! a_2! \dots a_n!, \\ \mathbf{b}^{\mathbf{a}} &= b_1^{a_1} b_2^{a_2} \dots b_n^{a_n}, \end{aligned}$$

where $\mathbf{a} = (a_1, a_2, \dots, a_n)$ and $\mathbf{b} = (b_1, b_2, \dots, b_n)$.

Remark: Our two-layer queueing network doesn't take into account pipelining for packet switching and retransmission mechanism of TCP. In terms of pipelining, a store-and-forward procedure at each router in real physical networks is processed on a packet by packet basis. In the two-layer queueing network model, the transfer unit is a file. This implies that the file transfer process at an intermediate link begins just after the file transfer process at the previous link is completed. This lack of pipelining effect causes a large delay in our model. On the other hand, retransmission mechanism of TCP causes an additional load in the physical network, and the degradation of the end-to-end file transfer delay is expected when the offered load is large. We will discuss on the issue in the subsection 5.1.

4 Analysis of two-layer queueing network

In this section, we apply the results of [5] to our two-layer queueing network model, and derive the end-to-end file transfer delay.

The traffic equation for a type- m file to visit stations in the upper layer is given by

$$(1, \mathbf{v}(m)) = (1, \mathbf{v}(m))\mathbf{S}(m), \quad (3)$$

where $\mathbf{v}(m) = (v_1(m), v_2(m), \dots, v_{N_I}(m))$, and $v_i(m)$ is the relative frequency with which a type- m file visits station i . Note that $v_i(m) = 0$ for $i \notin G_m$.

Similarly, the traffic equation for a station- i file to join queues in the lower layer is yielded as

$$(1, \mathbf{w}(i)) = (1, \mathbf{w}(i))\mathbf{R}(i), \quad (4)$$

where $\mathbf{w}(i) = (w_1(i), w_2(i), \dots, w_{N_J}(i))$ and $w_j(i)$ is the relative frequency with which a station- i file visits queue j . Note that $w_j(i) = 0$ for $j \notin H_i$. We also have

$$\sum_{i=1}^{N_I} v_i(m) s_{i0}(m) = 1, \quad m = 1, 2, \dots, N_M, \quad (5)$$

and

$$\sum_{j=1}^{N_J} w_j(i) r_{j0}(i) = 1, \quad i = 1, 2, \dots, N_I. \quad (6)$$

In the following, a type- m file being in one of queues associated with station i is called a (m, i) -file. From equations (3) to (6), the relative frequency with which a (m, i) -file visits queue j in the lower layer is given by

$$\theta_{j(m,i)} = v_i(m) w_j(i), \quad m = 1, 2, \dots, N_M, \quad i = 1, 2, \dots, N_I, \quad j = 1, 2, \dots, N_J.$$

With $\theta_{j(m,i)}$, the traffic intensity resulting from a (m, i) -file $\rho_{j(m,i)}$ is yielded as

$$\rho_{j(m,i)} = \frac{\theta_{j(m,i)}}{\mu_j}, \quad m = 1, 2, \dots, N_M, \quad i = 1, 2, \dots, N_I, \quad j = 1, 2, \dots, N_J.$$

We introduce the following notations in terms of $\rho_{j(m,i)}$.

$$\begin{aligned} \boldsymbol{\rho}_j(m) &= (\rho_{j(m,1)}, \rho_{j(m,2)}, \dots, \rho_{j(m,N_I)}), \\ \boldsymbol{\rho}_j &= (\boldsymbol{\rho}_{j(1)}, \boldsymbol{\rho}_{j(2)}, \dots, \boldsymbol{\rho}_{j(N_M)}), \\ \boldsymbol{\rho} &= (\boldsymbol{\rho}_1, \boldsymbol{\rho}_2, \dots, \boldsymbol{\rho}_{N_J}). \end{aligned}$$

Let $x_{j(m,i)}$ be the number of (m, i) -files at queue j , and denote

$$\begin{aligned} \mathbf{x}_j(m) &= (x_{j(m,1)}, x_{j(m,2)}, \dots, x_{j(m,N_I)}), \\ \mathbf{x}_j &= (\mathbf{x}_{j(1)}, \mathbf{x}_{j(2)}, \dots, \mathbf{x}_{j(N_M)}), \\ \mathbf{x} &= (\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_{N_J}). \end{aligned}$$

For convenience, we write $\mathbf{x} \geq \mathbf{0}$ if all elements in \mathbf{x} are greater than or equal to 0.

In our setting, the equilibrium marginal distribution for the aggregate state \mathbf{x} is given in the form

$$P(\mathbf{x}) = C\Lambda(\mathbf{x})\Phi(\mathbf{x}) \prod_{j=1}^{N_J} \frac{\|\mathbf{x}_j\|!}{\mathbf{x}_j!} \boldsymbol{\rho}_j^{\mathbf{x}_j},$$

where C is a normalization constant, $\Lambda(\mathbf{x})$ the arrival function, and $\Phi(\mathbf{x})$ the service-rate function [5]. Let $\boldsymbol{\lambda}_m$ ($m = 1, 2, \dots, N_M$) denote a $1 \times N_I$ vector whose elements are all the same and equal to λ_m . We define $\boldsymbol{\lambda}$ as

$$\boldsymbol{\lambda} = (\boldsymbol{\lambda}_1, \boldsymbol{\lambda}_2, \dots, \boldsymbol{\lambda}_{N_M}).$$

Then, $\Lambda(\mathbf{x})$ and $\Phi(\mathbf{x})$ are given by

$$\begin{aligned} \Lambda(\mathbf{x}) &= \boldsymbol{\lambda}^{\mathbf{x}_j}, \quad j = 1, 2, \dots, N_J, \\ \Phi(\mathbf{x}) &= 1. \end{aligned}$$

Taking the sum of $P(\mathbf{x})$ for all $\mathbf{x} \geq \mathbf{0}$ yields

$$\begin{aligned} \sum_{\mathbf{x} \geq \mathbf{0}} P(\mathbf{x}) &= C \prod_{j=1}^{N_J} \sum_{n=0}^{\infty} \sum_{\|\mathbf{x}_j\|=n} \frac{\|\mathbf{x}_j\|!}{\mathbf{x}_j!} \boldsymbol{\rho}_j^{\mathbf{x}_j} \boldsymbol{\lambda}^{\mathbf{x}_j} \\ &= C \prod_{j=1}^{N_J} \sum_{n=0}^{\infty} (\boldsymbol{\lambda}, \boldsymbol{\rho}_j)^n, \end{aligned}$$

where $(\boldsymbol{\lambda}, \boldsymbol{\rho}_j)$ is the inner product of $\boldsymbol{\lambda}$ and $\boldsymbol{\rho}_j$. We then obtain

$$C = \prod_{j=1}^{N_J} C_j, \tag{7}$$

where $C_j = 1 - (\boldsymbol{\lambda}, \boldsymbol{\rho}_j)$. Note that the necessary and sufficient condition for system stability is

$$(\boldsymbol{\lambda}, \boldsymbol{\rho}_j) < 1, \quad 1 \leq j \leq N_J.$$

Let $\phi_{j(m,i)}$ denote the average number of (m, i) -files at queue j . We obtain

$$\phi_{j(m,i)} = C_j \lambda_m \rho_{j(m,i)} \sum_{n=1}^{\infty} n (\boldsymbol{\lambda}, \boldsymbol{\rho}_j)^{n-1},$$

and hence from (7),

$$\phi_{j(m,i)} = \frac{\lambda_m \rho_{j(m,i)}}{1 - (\boldsymbol{\lambda}, \boldsymbol{\rho}_j)}.$$

Let $\tau_{j(m,i)}$ denote the throughput of (m, i) -files at queue j . Then, we obtain

$$\tau_{j(m,i)} = \lambda_m \theta_{j(m,i)}.$$

Using Little's formula, the average time of a (m, i) -file spent in queue j , denoted by $\eta_{j(m,i)}$, is given by

$$\eta_{j(m,i)} = \frac{\phi_{j(m,i)}}{\tau_{j(m,i)}}.$$

Then, the average time of a type- m file spent in station i , $T_{(m,i)}$, is yielded as

$$T_{(m,i)} = \sum_{j=1}^{N_J} w_j(i) \eta_{j(m,i)}.$$

With $T_{(m,i)}$, we obtain the end-to-end file transfer delay as

$$E[T] = \sum_{m=1}^{N_M} \sum_{i=1}^{N_I} \frac{\lambda_m}{\sum_{n=1}^{N_M} \lambda_n} v_i(m) T_{(m,i)}.$$

5 Numerical examples

In this section, we show some numerical examples of the mean end-to-end file-transfer delay for four-terminal four-router networks.

We consider the case in which one terminal transfers a requested file to the other terminal. A file request arrives at each terminal according to a Poisson process with rate λ , that is, $\lambda_m = \lambda$ for all m . The destination terminal of the file is equally likely, and its end-to-end path in the logical network is determined such that the number of hops along the path is minimum. If the number of paths with the minimum hop-number is $n (> 1)$, one of the paths is selected with probability $1/n$.

We assume that the average file size is f MB, and that the bandwidth of queue j ($j = 1, 2, \dots, N_J$) in the physical network is c_j Mbps. In this case, the service rate of queue j in the lower layer is given by $\mu_j = c_j/(8f)$. In the following, we

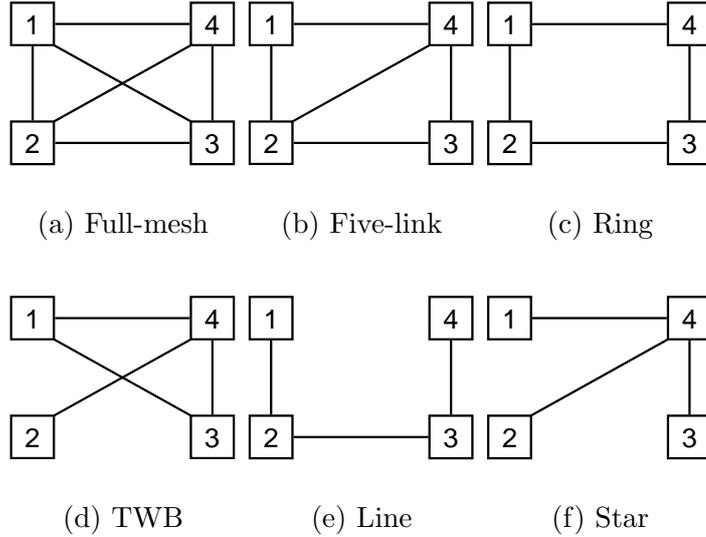


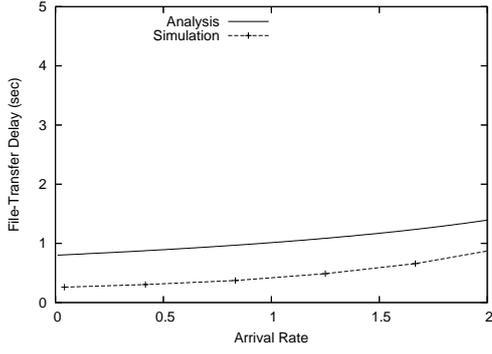
Fig. 5. Logical network topologies.

set $f = 3.0$ [MB] and $c = 100$ [Mbps]. We consider the six physical topologies shown in Fig. 3 and six logical topologies presented in Fig. 5.

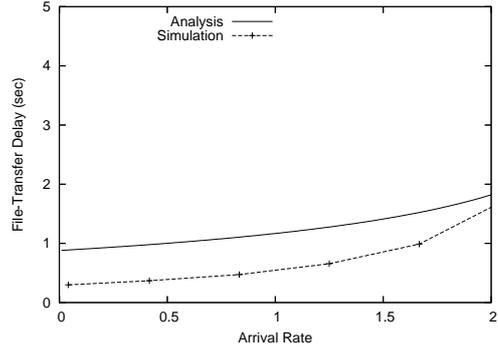
5.1 Model validation

We conducted ns-2 simulation experiments to validate the analysis model. In ns-2 simulation, file requests arrive at each terminal according to a Poisson process with rate λ . A file is transferred from its source overlay node by TCP protocol. A store-and-forward procedure at each overlay node is processed on a file by file basis. That is, a file is transferred from an overlay node, say A, to the next overlay node, say B, just after all packets comprising the file are arrived at overlay node A. If overlay node A has files to be transferred to overlay node B, those are transferred simultaneously. Note that in the physical network, a store-and-forward procedure at each router is processed on a packet by packet basis. In terms of the file size, exponential and Pareto distributions are considered, and the mean file size is 3.0 MB for both cases. (This is a typical size for audio file.) In Pareto distribution case, the shape parameter is set to 1.5. Note that the variance of the Pareto distribution case is infinity.

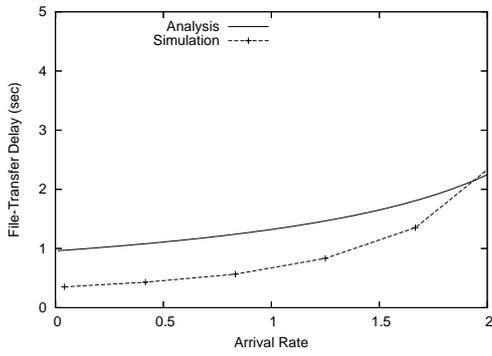
Figures 6(a) to 6(f) show the end-to-end file-transfer delay against the arrival rate λ . The file-size distribution in simulation is exponential. Here, the physical network topology is ring, and the end-to-end file-transfer delays for six overlay topologies are depicted. In Figs. 6(d), 6(e), and 6(f), simulation results are smaller than analysis ones when the arrival rate is small. Note that in simulation, a store-and-forward procedure at each router in the physical network is processed on a packet by packet basis, while in the analysis, the



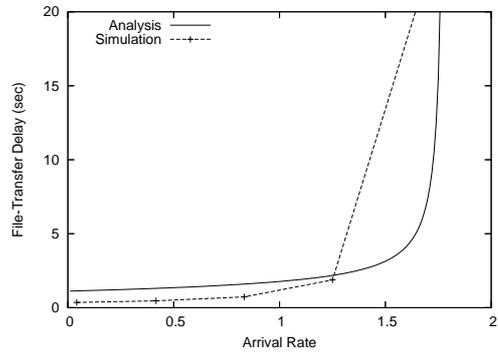
(a) Logical topology: Full-mesh.



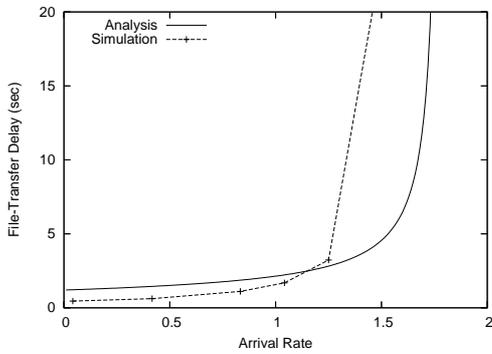
(b) Logical topology: Five-link.



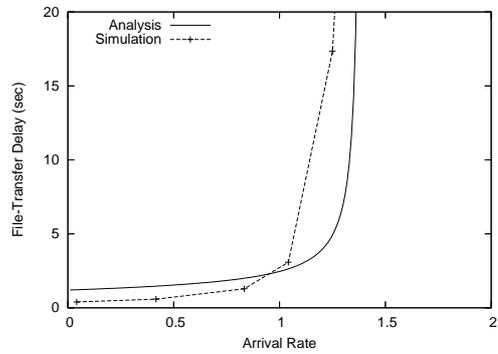
(c) Logical topology: Ring.



(d) Logical topology: TWB.



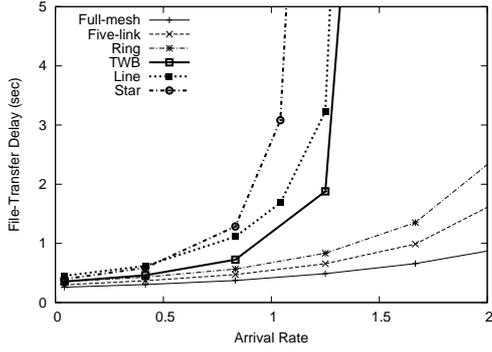
(e) Logical topology: Line.



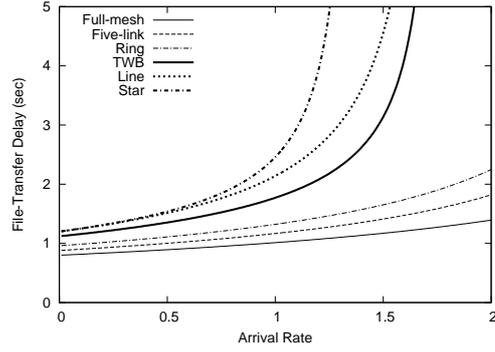
(f) Logical topology: Star.

Fig. 6. File-transfer delay. (Physical topology: Ring)

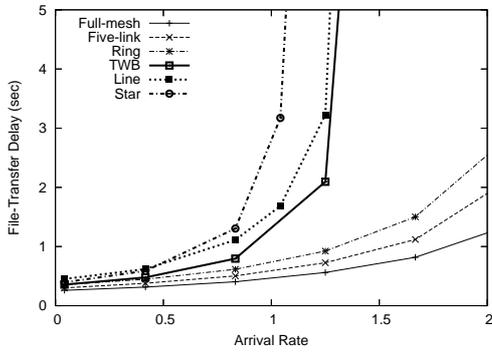
transfer unit in the physical network is a file (the lack of packet-level pipelining). This implies that the file transfer process at an intermediate link begins just after the file transfer process at the previous link is completed. In other words, the file transfer job in the analysis takes more time than that in the simulation. The same observation can be made in Figs. 6(a), 6(b) and 6(c),



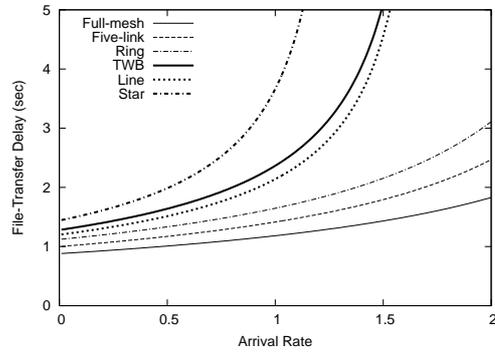
(a) Simulation (Physical topology: Ring)



(b) Analysis (Physical topology: Ring)



(c) Simulation (Physical topology: Line)



(d) Analysis (Physical topology: Line)

Fig. 7. Comparison of Simulation and Analysis.

where simulation results are smaller than analysis results for any λ .

Figures 6(d), 6(e), and 6(f) also show that simulation results are rapidly increasing and greater than analysis ones when the arrival rate is large. This performance degradation is caused by TCP. We observed from simulation data that packet loss frequently occurs at a bottleneck router when λ is large. This frequent packet loss makes the TCP transmission rate small, as well as generating a large amount of retransmission traffic. We also observed this performance degradation in cases of Figs. 6(a), 6(b) and 6(c) when the arrival rate is greater than two. This result implies that when physical and logical topologies are consistent, congestion is not likely to occur. In other words, the end-to-end file-transfer delay in topology consistent case is less sensitive to the traffic intensity than topology inconsistent case.

Next, we investigate how the topology inconsistency affects the mean end-to-end file-transfer delay, comparing analysis with simulation. Figures 7 show

Table 2

Relative difference of file-transfer delays between exponential and deterministic distributions (10^{-2}).

Logical topology	$\lambda = 0.0417$	$\lambda = 0.4167$	$\lambda = 0.8333$	$\lambda = 1.25$
Full-mesh	3.19	3.46	1.90	3.41
Five-link	2.52	3.15	1.87	2.11
Ring	4.57	3.57	2.44	3.51
TWB	5.30	4.70	4.07	16.9
Line	3.12	3.98	4.91	13.2
Star	4.45	5.67	4.96	19.5

Table 3

Relative difference of file-transfer delays between exponential and Pareto distributions (10^{-2}).

Logical topology	$\lambda = 0.0417$	$\lambda = 0.4167$	$\lambda = 0.8333$	$\lambda = 1.25$
Full-mesh	14.7	0.15	3.38	5.28
Five-link	15.9	3.60	0.38	2.84
Ring	11.3	5.69	0.15	2.22
TWB	13.4	0.59	5.59	18.8
Line	16.6	3.01	2.35	3.43
Star	14.5	5.78	10.8	42.6

the mean end-to-end file-transfer delay against λ . The physical topology of Figs. 7(a) and 7(b) is ring, while that of Figs. 7(c) and 7(d) is line. Figures 7(a) and 7(c) show simulation results, and Figs. 7(b) and 7(d) represent analysis ones.

In the ring physical topology case, we observe the quantitative difference between Figs. 7(a) and 7(b), however, the change of the end-to-end file-transfer delay in terms of logical topologies for simulation is the same as that for analysis. We observed the same qualitative nature when the physical topology is full-mesh, five-link, TWB and star. The only exception is the case of line physical topology. In Figs. 7(c) and 7(d), the analysis results of TWB and line logical topologies exhibit the different qualitative tendency from the simulation ones. Note that the simulation results of the other four logical topologies exhibit the same tendency as the analysis results. That is, simulation and analysis results are qualitatively consistent for 34 of 36 combinations of logical and physical topologies. This suggests that the analysis is efficient in a qualitative sense to investigate the effect of the topology inconsistency on the end-to-end file-transfer delay.

Table 4

Comparison of file-transfer delay for ring and TWB logical topologies.

Physical topology	Low arrival rate ($0 < \lambda < 0.8$)	High arrival rate ($\lambda > 1.2$)
Full-Mesh	$T_{ring} < T_{TWB}$	$T_{ring} < T_{TWB}$
Five-link	$T_{ring} < T_{TWB}$	$T_{ring} < T_{TWB}$
Ring	$T_{ring} < T_{TWB}$	$T_{ring} < T_{TWB}$
TWB	$T_{ring} > T_{TWB}$	$T_{ring} < T_{TWB}$
Line	$T_{ring} < T_{TWB}$	$T_{ring} < T_{TWB}$
Star	$T_{ring} > T_{TWB}$	$T_{ring} < T_{TWB}$

Finally, we investigate how the file size distribution affects the simulation results. Note that the file size is assumed to be generally distributed in the analysis. Therefore, the file size distribution does not affect the mean end-to-end file-transfer delay calculated from the analysis. Tables 2 and 3 show the relative difference of the end-to-end file-transfer delays between exponential and deterministic distributions and between exponential and Pareto distributions, respectively. The relative difference is the ratio of absolute difference of two delays to the exponential-distribution case. The physical topology considered here is ring.

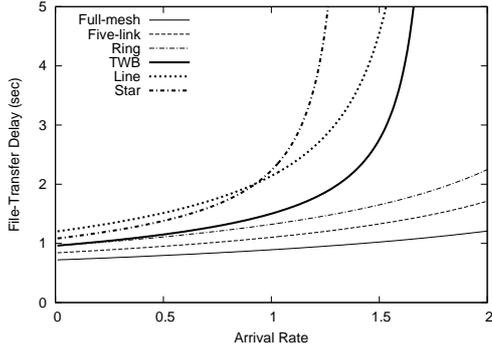
In both tables, when $\lambda = 1.25$, the star logical topology exhibits the worst value, resulting from congestion due to a large arrival rate. In other cases, however, the difference between two distributions is significantly small. This implies that the end-to-end file-transfer delay is insensitive to the file size distribution, validating the assumption that the file size is generally distributed.

5.2 Impact of topology inconsistency on file transfer delay

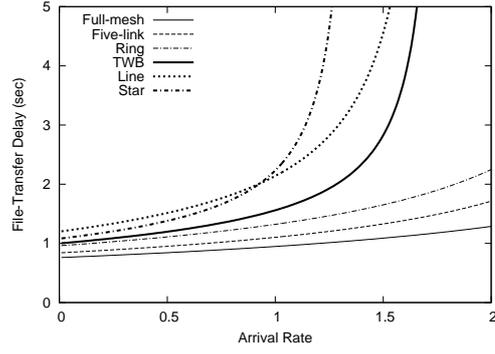
In this subsection, we show some numerical examples calculated from the analysis result, discussing how the topology inconsistency affects the end-to-end file transfer delay.

5.2.1 Impact of logical topology

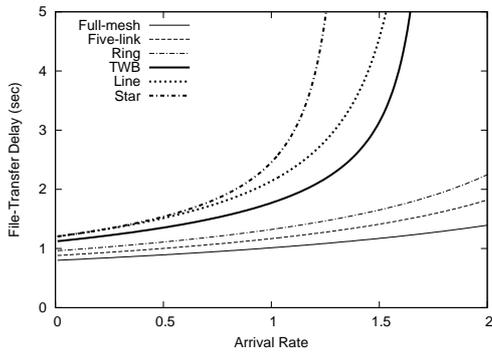
Figures 8(a) to 8(f) illustrate the end-to-end file-transfer delay against the arrival rate λ . In each figure, the end-to-end file-transfer delays for six logical topologies are compared over a physical network topology. In all the cases, the full-mesh logical topology achieves the smallest file-transfer delay, and the five-link logical topology provides the second smallest one. In all the cases except line physical topology, the ring and TWB logical topologies provide smaller



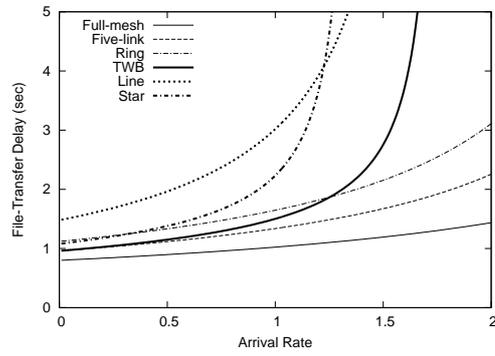
(a) Physical topology: Full-mesh



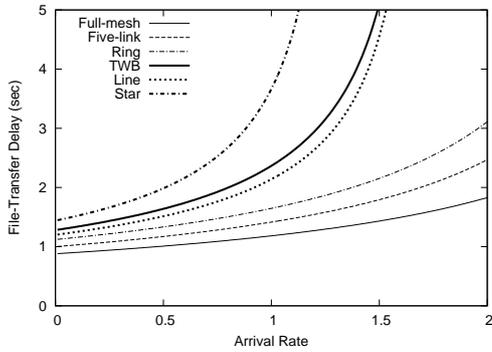
(b) Physical topology: Five-link



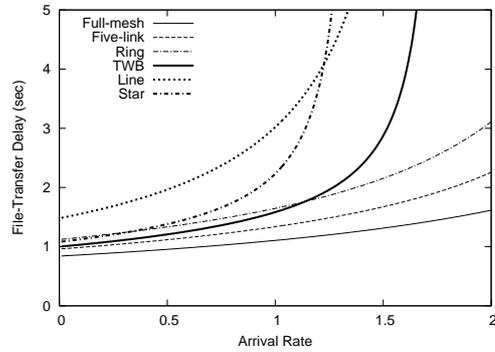
(c) Physical topology: Ring



(d) Physical topology: TWB



(e) Physical topology: Line



(f) Physical topology: Star

Fig. 8. File-transfer delay for six physical topologies.

file-transfer delay than the line and star topologies. This is simply because a small mean number of hops (physical links) between source and destination pair of overlay nodes makes the end-to-end file-transfer delay small.

Table 4 (Table 5) shows the impact of topology inconsistency on the end-to-

Table 5

Comparison of file-transfer delay for line and star logical topologies.

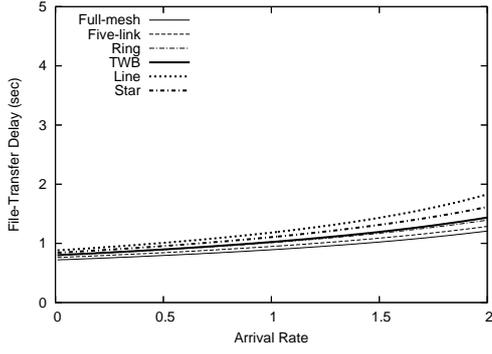
Physical topology	Low arrival rate ($0 < \lambda < 0.8$)	High arrival rate ($\lambda > 1.2$)
Full-Mesh	$T_{line} > T_{star}$	$T_{line} < T_{star}$
Five-link	$T_{line} > T_{star}$	$T_{line} < T_{star}$
Ring	$T_{line} = T_{star}$	$T_{line} < T_{star}$
TWB	$T_{line} > T_{star}$	$T_{line} < T_{star}$
Line	$T_{line} < T_{star}$	$T_{line} < T_{star}$
Star	$T_{line} > T_{star}$	$T_{line} < T_{star}$

end file-transfer delay in terms of ring and TWB topologies (line and star topologies). In these tables, T_G represents the end-to-end file-transfer delay for G -type logical topology. When the arrival rate is large, Table 4 shows that the end-to-end file-transfer delay for the TWB topology is larger than that for the ring one, and Table 5 represents that the end-to-end file-transfer delay for star topology is larger than that for the line one. In the star logical topology, most of requested files traverses the center overlay node and hence the center overlay node is likely to be congested, resulting in the degradation of network performance. Similarly, when the logical topology is TWB, the node whose degree is three is more likely to be congested in comparison with the nodes of the ring logical topology. Therefore, the network performance of the TWB logical topology is worse than that of the ring one.

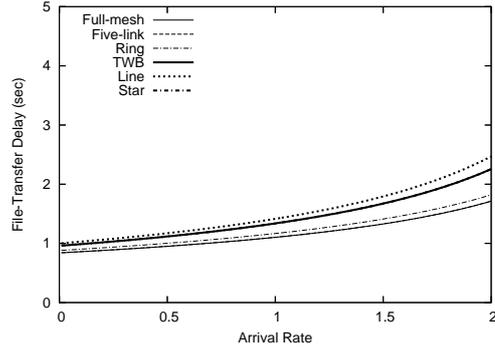
When the arrival rate is small, on the other hand, the end-to-end file-transfer delay in the case where both logical and physical topologies are the same is smaller than that in the case where both topologies are different. (See the results of ring and TWB physical topologies for a low arrival rate in Table 4 and the results of line and star physical topologies in Table 5.) In terms of the ring physical network, T_{line} is equal to T_{star} . This is because the mean number of hops between source and destination pair of overlay nodes for the line topology is equal to that for the star one. When the physical network is one of full-mesh, five-link and line, T_{TWB} is larger than T_{ring} . This is because the mean number of physical links between source and destination pair of overlay nodes for the TWB logical topology is greater than that for the ring logical topology. The same reason applies to the following cases: (1) T_{ring} is larger than T_{TWB} when the physical network is star, (2) T_{line} is larger than T_{star} when the physical network is one of full-mesh, five-link, and TWB.

5.2.2 Impact of physical topology

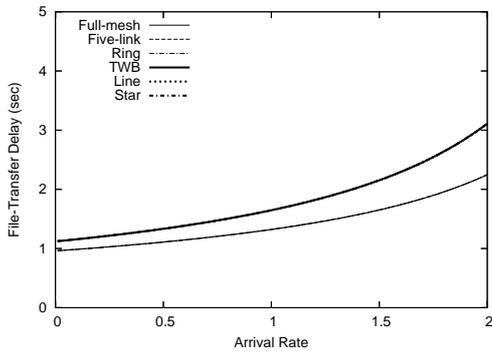
Figures 9(a) to 9(f) show the end-to-end file-transfer delay against the arrival rate λ . In each figure, the end-to-end file-transfer delays for six physical



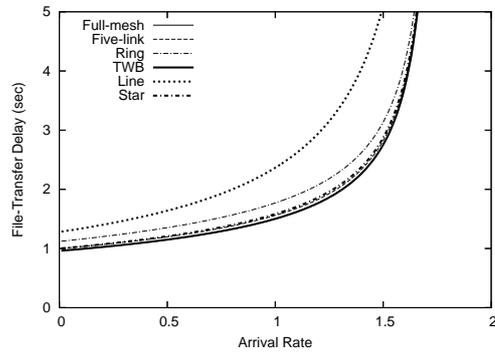
(a) Logical topology: Full-mesh



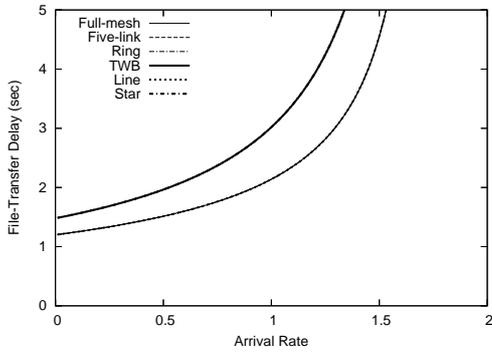
(b) Logical topology: Five-link



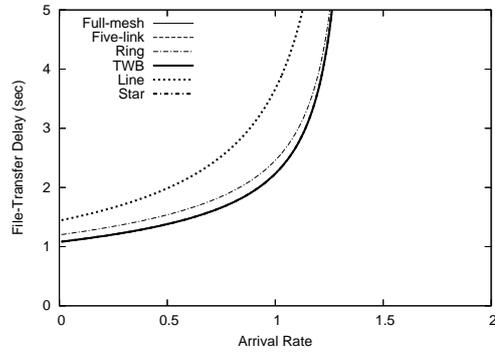
(c) Logical topology: Ring



(d) Logical topology: TWB



(e) Logical topology: Line



(f) Logical topology: Star

Fig. 9. File-transfer delay for six logical topologies. The overlapping curves are: (b) full-mesh with five-link, TWB with star, (c) full-mesh with five-link and ring, TWB with line and star, (d) TWB with full-mesh, (e) full-mesh with five-link, ring and line, TWB with star, and (f) TWB with full-mesh, five-link and star.

topologies are compared over a logical network topology. In Fig. 9(b), when the logical topology is five-link, the end-to-end file-transfer delays for the physical

topologies of full-mesh and five-link are the same. This is because when the logical topology is five-link, the physical links used for file transfer in those physical topologies are the same. The same tendency is observed in the following cases: (1) the logical topology is five-link and the physical topology is star or TWB (Fig. 9(b)), (2) the logical topology is ring and the physical topology is one of full-mesh, five-link, and ring (Fig. 9(c)), (3) the logical topology is ring and the physical topology is TWB or star (Fig. 9(c)), (4) the logical topology is TWB and the physical topology is full-mesh or TWB (Fig. 9(d)), (5) the logical topology is line and the physical topology is one of full-mesh, five-link, ring, and line (Fig. 9(e)), (6) the logical topology is line and the physical topology is TWB or star (Fig. 9(e)), and (7) the logical topology is star and the physical topology is one of full-mesh, five-link, TWB, and star (Fig. 9(f)). We also observe from Fig. 9(c) that when the logical topology is ring, the end-to-end file-transfer delays for line and star physical topologies are the same.

From Figs. 9(a) to 9(f), it is observed that the end-to-end file-transfer delay is small in the following two cases: (1) the physical topology is consistent with the logical topology, and (2) the physical topology completely covers the logical topology. Focusing on topology inconsistent cases in Fig. 9(a), we also observe that when the logical topology is full-mesh, the end-to-end file-transfer delay for the five-link physical network is the smallest and that for the line one the largest, except for the full-mesh physical network. This is because the physical topology whose mean number of physical links between source and destination pair of overlay nodes is small provides a small end-to-end file-transfer delay.

In Figs. 9(d), 9(e), and 9(f), the end-to-end file-transfer delay increases rapidly against λ for all the six physical topologies, on the other hand, in Figs. 9(a), 9(b), and 9(c), a gradual increase in delay is observed. When the logical topology is line, nodes 2 and 3 are frequently used for file transmission, that is, these are bottleneck overlay nodes. The exponential increase in the end-to-end file-transfer delay results from the bottleneck overlay nodes. This is also true in case of the TWB and star logical topologies, as shown in Figs. 9(d) and 9(f) respectively. When the logical topology is five-link (Fig. 9(b)) or ring (Fig. 9(c)), on the other hand, the traffic load is equally distributed to all overlay nodes, resulting in a small end-to-end file-transfer delay. Note that the full-mesh logical topology provides a small end-to-end file-transfer delay for the six physical topologies.

Note that for each figure in Fig. 9, all the end-to-end file-transfer delays change in a similar way. In Figs. 8(a) to 8(f), on the other hand, the difference among the curves is significantly large. This implies that for the four-terminal four-router network, the end-to-end file-transfer delay is greatly affected by logical topology, rather than physical topology.

Table 6
Mean number of physical-layer hops.

Physical topology	Logical topology					
	Full-mesh	Five-link	Ring	TWB	Line	Star
Full-mesh	3.000	3.500	4.000	4.000	5.000	4.500
Five-link	3.167	3.500	4.000	4.167	5.000	4.500
Ring	3.333	3.667	4.000	4.667	5.000	5.000
TWB	3.333	4.000	4.667	4.000	6.167	4.500
Line	3.667	4.167	4.667	5.333	5.000	6.000
Star	3.500	4.000	4.667	4.167	6.167	4.500

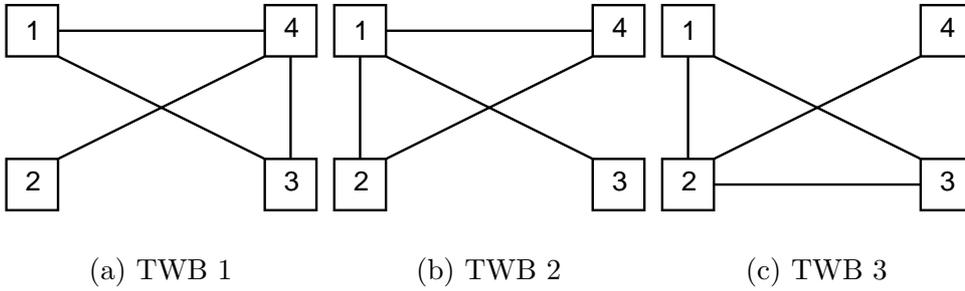


Fig. 10. Three TWB logical topologies.

In order to clarify the reason, we consider the mean numbers of physical-layer hops for all the combinations of logical and physical topologies. Table 6 represents the mean number of physical-layer hops from source to destination. It is observed from this table that when a physical (logical) topology is fixed, the mean number of physical-layer hops changes according to logical (physical) topology. A remarkable point is that the degree of variation by logical topology is greater than that by physical topology. For example, when physical layer is full-mesh, the variation range is from 3.000 to 5.000. (See full-mesh row in Table 6.) When logical layer is fixed with full-mesh, on the other hand, the variation range is from 3.000 to 3.667. (See full-mesh column in Table 6.) This small degree of variation for full-mesh logical network makes the end-to-end file-transfer delay less sensitive to physical topology, as shown in Fig. 9(a).

5.2.3 Same topology-type case

We consider the topology inconsistency case in which physical and logical topologies are not the same, but they belong to the same topology category. We focus on TWB type for both logical and physical topologies. We choose Fig. 3(d) as a physical TWB topology, while three TWB types shown in Fig. 10

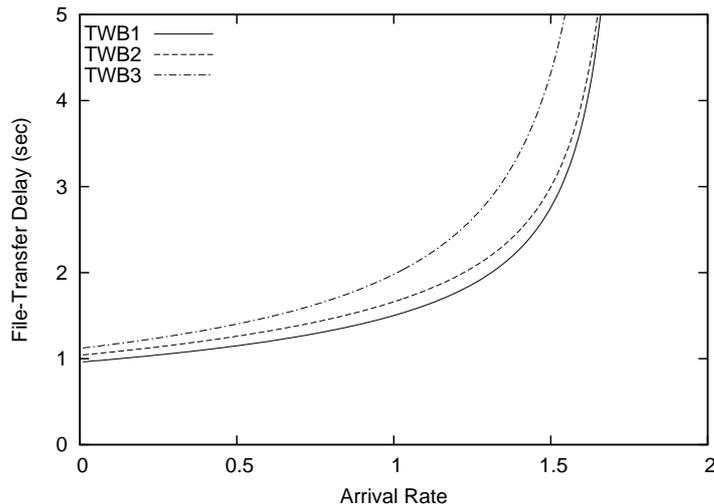


Fig. 11. File-transfer delay for three TWB logical topologies. (Physical topology: TWB)

are considered for logical topology. Figure 11 represents the end-to-end file-transfer delay against λ for three TWB logical topologies. It is observed from Fig. 11 that the end-to-end file-transfer delay for TWB 1 case is the smallest, while that for TWB 3 case is the largest. This is because TWB 1 is completely consistent with the physical topology, while TWB 3 is the most inconsistent case. Note that the discrepancy among three cases is relatively small in comparison with that among different logical topology types, as shown in Fig. 8(d). This result suggests that the end-to-end file transfer delay does not change significantly among the same logical topology class.

5.2.4 Performance improvement by bandwidth enhancement

Consider the network in which the logical and physical networks are line topology. The queues of the physical network are classified into two types: queues between a terminal and a router, and those between two routers. We call the former access-link queues and the latter core-link queues. Let c_a and c_c denote the bandwidth of an access-link queue and that of a core-link queue, respectively.

Figure 12 shows the end-to-end file transfer delay against the arrival rate. We consider the following three cases for the bandwidths of access- and core-link queues: $(c_a, c_c) = (50, 50)$, $(50, 100)$ and $(100, 100)$ ($([Mbps], [Mbps])$). It is observed from the figure that the file-transfer delays in cases of $(c_a, c_c) = (50, 50)$ and $(50, 100)$ increase rapidly. This is simply because the bandwidth of access-link queues is small. We also observe that the end-to-end file-transfer delay for $(c_a, c_c) = (100, 100)$ is significantly smaller than those for the other two cases. This result implies that the end-to-end file-transfer delay is greatly

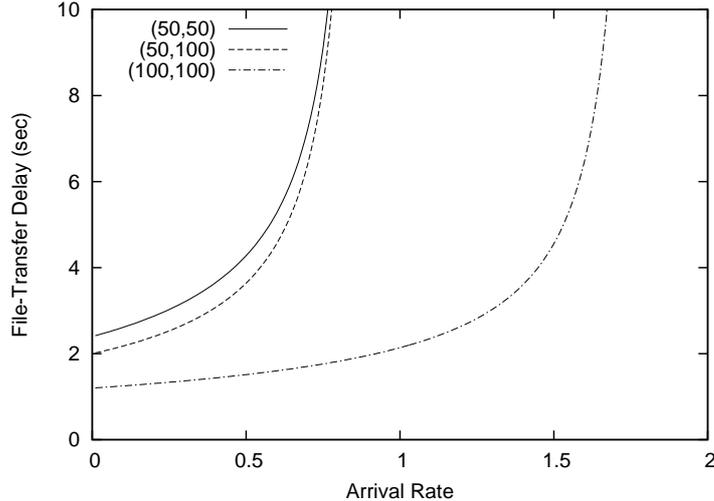


Fig. 12. Effect of link bandwidth on file-transfer delay.

affected by the bandwidth of access-link queues.

6 Conclusion

In this paper, focusing on six types of four-terminal four-router network, we analyzed the mean end-to-end file-transfer delay by using two-layer queueing network model. The analysis has been validated in a qualitative sense by ns-2 simulation. Numerical examples showed that the full-mesh logical topology achieves the smallest file-transfer delay in six types of logical topologies. It was also presented that ring, TWB, line, and star logical topologies provide different performance due to physical topologies. In particular, we found that when both logical and physical topologies are the same, the end-to-end file-transfer delay is significantly small. We also illustrated that the end-to-end file-transfer delay depends on the traffic intensity for TWB, line, and star logical topologies, regardless of physical topology. Furthermore, it was verified that the enhancement of access-link bandwidth can greatly improve the end-to-end file-transfer delay.

A remarkable point observed from the numerical results is that for the four-terminal four-router network, the end-to-end file-transfer delay is greatly affected by logical topology, rather than physical topology. This is because the mean number of physical-layer hops varies greatly according to logical topology.

It is significant to note that the consistent mapping between logical and physical topologies does not necessarily provide a low file-transfer delay. Our numerical examples showed that full-mesh logical topology always gives the lowest

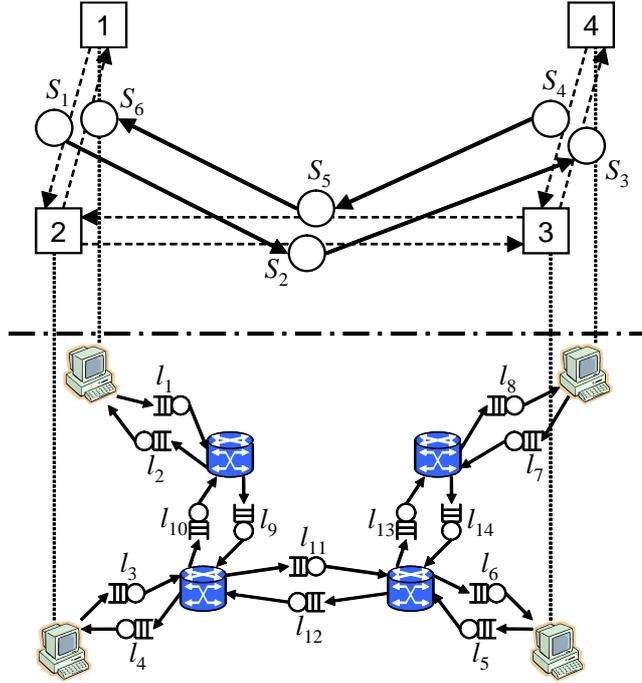


Fig. A.1. Example of a two-layer queueing network

end-to-end file-transfer delay over any physical topology. This implies that good connectivity for overlay level is also important for achieving a small transfer delay.

Note that these result are obtained in the four-terminal four-router network. In order to validate the effectiveness of overlay networking, further study is needed for large-sized networks.

Acknowledgments

The authors would like to thank Ms. Kyoko Ashitagawa of NTT Advanced Technology for assistance in the simulation experiments. They are grateful to the anonymous reviewers for many valuable comments. This research was supported in part by the International Communications Foundation (ICF).

A Example of Markovian routing chain

In this appendix, we present an example of Markovian routing chains $\mathbf{S}(m)$ of (1) and $\mathbf{R}(i)$ of (2) for the two-layer queueing network shown in Fig. A.1.

First, consider the case where a file is transferred from overlay node 1 to the

other node. The file is transferred to node 2 first, that is, the file traverses through station S_1 first. This gives $s_{01}(1) = 1$. When the file reaches to node 2, the next event is the following two cases: (1) the file leaves the network at node 2, (2) the file moves to node 3 through station S_2 . Because the destination terminal of the file is equally likely, we have $s_{10}(1) = 1/3$ and $s_{12}(1) = 2/3$. If the file moves to node 3, the file leaves the network or moves to node 4. Therefore, we obtain $s_{20}(1) = 1/2$ and $s_{23}(1) = 1/2$. Finally, if the file arrives at node 4, the file leaves the network with probability one, and hence $s_{30}(1) = 1$. Therefore, we have the followings for $\mathbf{S}(1)$:

$$s_{ij}(1) = \begin{cases} 1, & (i, j) = (0, 1) \text{ and } (3, 0), \\ 2/3, & (i, j) = (1, 2), \\ 1/2, & (i, j) = (2, 0) \text{ and } (2, 3), \\ 1/3, & (i, j) = (1, 0), \\ 0, & \text{otherwise.} \end{cases}$$

Similarly, we obtain

$$s_{ij}(2) = \begin{cases} 1, & (i, j) = (3, 0) \text{ and } (6, 0), \\ 2/3, & (i, j) = (0, 2), \\ 1/2, & (i, j) = (2, 0) \text{ and } (2, 3), \\ 1/3, & (i, j) = (0, 6), \\ 0, & \text{otherwise,} \end{cases}$$

$$s_{ij}(3) = \begin{cases} 1, & (i, j) = (3, 0) \text{ and } (6, 0), \\ 2/3, & (i, j) = (0, 5), \\ 1/2, & (i, j) = (5, 0) \text{ and } (5, 6), \\ 1/3, & (i, j) = (0, 3), \\ 0, & \text{otherwise,} \end{cases}$$

$$s_{ij}(4) = \begin{cases} 1, & (i, j) = (0, 4) \text{ and } (6, 0), \\ 2/3, & (i, j) = (4, 5), \\ 1/2, & (i, j) = (5, 0) \text{ and } (5, 6), \\ 1/3, & (i, j) = (4, 0), \\ 0, & \text{otherwise.} \end{cases}$$

In terms of $\mathbf{R}(i)$, note that station S_1 consists of physical links l_1 , l_9 and l_4 . The file traverses through l_1 , l_9 , and l_4 . Therefore, we obtain

$$r_{ij}(1) = \begin{cases} 1, & (i, j) = (0, 1), (1, 9), (9, 4), \text{ and } (4, 0), \\ 0, & \text{otherwise.} \end{cases}$$

We similarly obtain

$$r_{ij}(2) = \begin{cases} 1, & (i, j) = (0, 3), (3, 11), (11, 6), \text{ and } (6, 0), \\ 0, & \text{otherwise,} \end{cases}$$

$$r_{ij}(3) = \begin{cases} 1, & (i, j) = (0, 5), (5, 13), (13, 8), \text{ and } (8, 0), \\ 0, & \text{otherwise,} \end{cases}$$

$$r_{ij}(4) = \begin{cases} 1, & (i, j) = (0, 7), (7, 14), (14, 6), \text{ and } (6, 0), \\ 0, & \text{otherwise,} \end{cases}$$

$$r_{ij}(5) = \begin{cases} 1, & (i, j) = (0, 5), (5, 12), (12, 4), \text{ and } (4, 0), \\ 0, & \text{otherwise,} \end{cases}$$

$$r_{ij}(6) = \begin{cases} 1, & (i, j) = (0, 3), (3, 10), (10, 2), \text{ and } (2, 0), \\ 0, & \text{otherwise.} \end{cases}$$

References

- [1] D. G. Andersen, H. Balakrishnan, F. Kaashoek and R. Morris, "Resilient Overlay Networks," *ACM SIGOPS*, vol. 35, no. 5, pp. 131–145, 2001.
- [2] Z. Duan, Z.-L. Zhang, and Y. T. Hou, "Service Overlay Networks: SLAs, QoS, and Bandwidth Provisioning," *IEEE/ACM Transactions on Networking*, vol. 11, no. 6, pp. 870–883, 2003.
- [3] Z. Ge, D. R. Figueiredo, S. Jaiswal, J. Kurose, and D. Towsley, "Modeling Peer-Peer File Sharing Systems," *INFOCOM 2003*, vol. 22, no. 1, pp. 2188–2198, 2003.
- [4] P. E. Green, "Fiber to the Home: The Next Big Broadband Thing," *IEEE Communications Magazine*, vol. 42, no. 9, pp. 100–106, 2004.

- [5] I. Kino, "Two-Layer Queueing Networks," *Journal of the Operations Research Society of Japan*, vol. 40, no. 2, pp. 163-185, 1997.
- [6] T. Kurasugi and I. Kino, "Approximation Methods for Two-Layer Queueing Models," *Performance Evaluation*, vol. 36, no. 1, pp. 55-70, 1999.
- [7] Z. Li and P. Mohapatra, "QRON: QoS-Aware Routing in Overlay Networks," *IEEE Journal on Selected Areas in Communications*, vol. 22, no. 1, pp. 29-40, 2004.
- [8] Z. Li and P. Mohapatra, "The Impact of Topology on Overlay Routing Service," *IEEE INFOCOM 2004*, 2004.
- [9] Y. Liu, H. Zhang, W. Gong and D. Towsley, "On the Interaction between Overlay Routing and Underlay Routing," *IEEE INFOCOM 2005*, vol. 4, pp. 2543-2553, 2005.
- [10] D. Qiu and R. Srikant, "Modeling and Performance Analysis of BitTorrent-Like Peer-to-Peer Networks," *ACM SIGCOMM 2004*, pp. 367-378, 2004.
- [11] K. K. Ramachandran and B. Sikdar, "An Analytic Framework for Modeling Peer to Peer Networks," *IEEE INFOCOM 2005*, vol. 3, pp. 2159-2169, 2005.
- [12] X. Yang and G. de Veciana, "Performance of Peer-to-Peer Networks: Service Capacity and Role of Resource Sharing Policies," *Performance Evaluation*, vol. 63, no. 3, pp. 175-194, 2006.
- [13] H. Zhang, J. Kurose and D. Towsley, "Can an Overlay Compensate for a Careless Underlay?," *IEEE INFOCOM 2006*, vol. 25, no. 1, pp. 2824-2835, 2006.