1993

Performance analysis of interconnected LANs with server/client configuration

Jiangling DU
National University of Singapore

Robert H. DENG
Singapore Management University, robertdeng@smu.edu.sg

Chi Chung KO
National University of Singapore

DOI: https://doi.org/10.1016/0169-7552(93)90022-v

Published in Computer Networks and ISDN Systems, Volume 25, Issue 12, July 1993, Pages 1321-1333.
https://doi.org/10.1016/0169-7552(93)90022-V

Performance analysis of interconnected LANs with server/client configuration

Jiangling Du, Robert H. Deng, Chi Chung Ko

Department of Electrical Engineering, National University of Singapore, Kent Ridge, Singapore 0511

Abstract

Du, J., R.H. Deng and C.C. Ko, Performance analysis of interconnected LANs with server/client configuration, Computer Networks and ISDN Systems 25 (1993) 1321–1333.

In this paper, we study the end-to-end performance of interconnected local area networks (LAN) with server/client configuration. The system uses bridges to connect two token-ring LANs through a high-speed communication link. A server station located on one LAN receives requests from client-stations on the same LAN as well as on the remote LAN, processes the requests, and returns responses to the client-stations. The end-to-end connections of the interconnected network are modelled as single-chain and multiple-chain closed queueing systems, which are solved by an iterative algorithm based on the MVA (mean value analysis) method. The performance examples are shown in terms of various system parameters such as the window size, server processing speed and internetwork transmission capacity, and are verified by computer simulations.

Keywords: Performance evaluation; token-ring LAN; interconnected LANs; mean value analysis.

1. Introduction

Local area networks (LAN) offer high-speed communication between distributed system components, such as workstations and shared resources; however, they are limited in geography and number of stations. With the large number of LANs now in use in universities and industry, interconnection of LANs over several to tens of kilometres is becoming common interests [4]. Thus, it is useful to study the performance of interconnected LANs in order to facilitate the design of such systems. Most of the studies of LANs and interconnected LANs were concerned with the system performance at the lower layers in the context of the OSI sever-layer reference model. However, a good performance at the lower layers is only the first step towards high-speed end-to-end communications. The overhead and processing delays of the higher layer protocols may significantly reduce the network transmission capacity that can be utilized by application programs. Recently, Murata and Takagi [6] built a two-layer performance model of a token-ring LAN. The performance model consists of a MAC (media access control) layer submodel and transport layer submodels.

In this paper, we generalize the study of [6] by considering interconnected LANs with server/client configuration. The interconnected network under investigation is depicted in Fig. 1. It uses bridges to connect two token-ring LANs through a high-speed communication link. A server station (server), which may be a database or a laser printer, is located on one LAN and is accessed by local as well as remote user stations (clients). There are two types of traffic in the interconnected system. Those messages which originate from one LAN and are destined for the other LAN are called inter-LAN messages; those which are transmitted over the same LAN are called intra-LAN messages. Bridges play an important role in handling the inter-LAN traffic. The basic

Correspondence to: J. Du, Department of Electrical Engineering, National University of Singapore, Kent Ridge, Singapore.

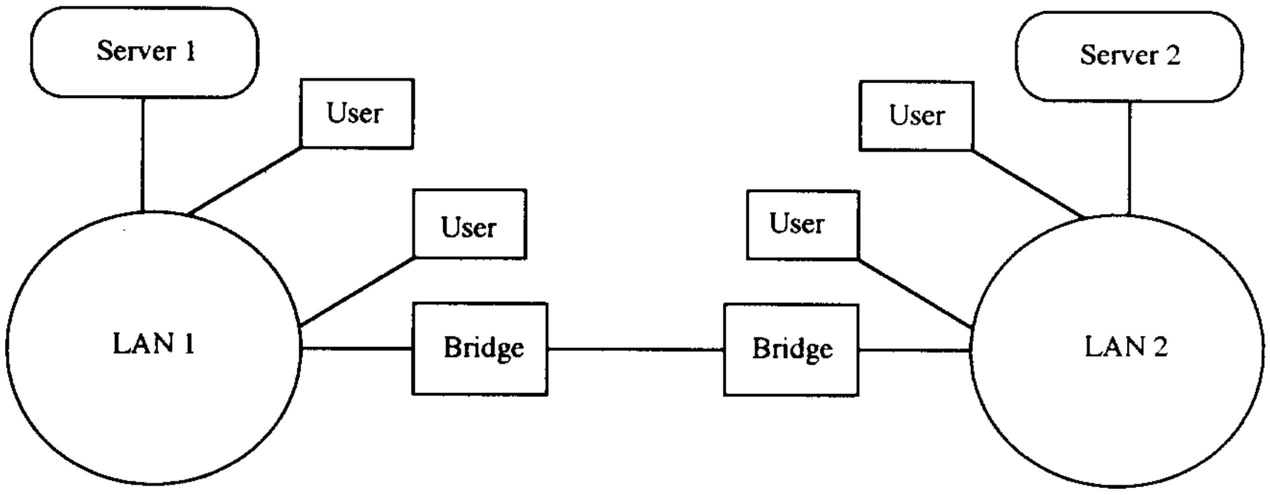
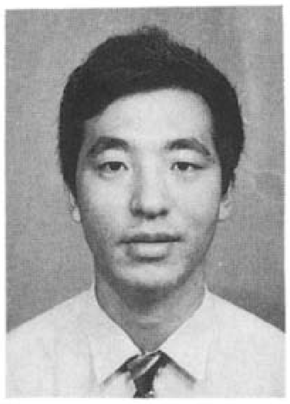


Fig. 1. An interconnected LAN system.

functions of the bridge are summarized in the following:

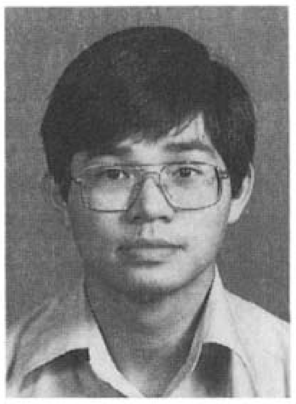
- It reads all successfully transmitted messages on the LAN it is attached to, and accepts all the inter-LAN messages whose final destination is on the other LAN.

Jiangling Du received his B.Sc., M.Sc. and Ph.D degrees in electrical engineering all from the Beijing Institute of Technology, Beijing, China, in 1983, 1986 and 1989, respectively. Between 1989 and 1991 he was a Research Staff member at LEI YIN Technology Corporation, Beijing, China. Currently he is a Research Assistant at the Department of Electrical Engineering, National University of Singapore. His research interests include adaptive signal design and processing for radar and communication systems, network interconnection, local area networks and performance evaluation.



Robert H. Deng received his B.E. degree from Changsha Institute of Technology, Changsha, China, in 1981 and the M.S. and Ph.D. degrees from the Illinois Institute of Technology, Chicago, in 1983 and 1985, respectively.

From 1986 and 1987 he was a Research Associate at the Department of Electrical and Computer Engineering, University of Notre Dame, IN, USA. Between 1987 and 1991 he was a Research Staff Member at the Institute of Systems Science, National University of Singapore, Singapore. Currently he is a Senior Lecturer at the Department of Electrical Engineering, National University of Singapore. His research interests include error-control coding techniques, trellis coded modulations, mobile radio communications, network interconnection, and network fault management.



Ko Chi Chung received the B.Sc. and Ph.D. degrees in electrical engineering from Loughborough University of Technology, UK in 1978 and 1981, respectively. In 1982, he joined the Department of Electrical Engineering, National University of Singapore, where he is presently a Senior Lecturer. His current research interests include adaptive arrays, digital signal processing and computer networks. He has written over 80 technical publications in these areas.

- It relays inter-LAN messages to the bridge of the other LAN via the bridge-to-bridge communication (internet) link.
- It broadcasts inter-LAN messages received from the remote bridge to the workstations of its own LAN.

The organization of the paper is as follows. In Section 2, we first outline the OSI layered protocol structure used in LAN and the queueing model of the token-passing MAC layer. We then present the end-to-end performance model of the interconnected network with server/client configuration. In Section 3, we develop a solution algorithm to the performance model of Section 2 based on the MVA method [7]. In Section 4, numerical results obtained from the analysis are shown, and are compared with simulations. The impacts of various system parameters on the end-to-end performance are discussed. Section 5 gives the conclusion of the study.

2. Internetwork end-to-end performance model

The layered protocol structure used in the internetwork is shown in Fig. 2. Application programs on workstations communicate with each other through three layers, i.e. the transport layer, the LLC (logical link control) layer and the MAC layer. The MAC layer employs a token-passing protocol with limited service. The LLC layer provides two types of services to the layer above it, connection oriented services and connectionless services. In this study we only consider connectionless service for the LLC layer. The connectionless service is a datagram-style of service. It simply allows for sending and receiving fully addressed datagrams, with no form of acknowledgement to assure reliable delivery. The responsibilities of error recovery, flow control and resequencing of messages are left to the connection oriented transport layer protocol.

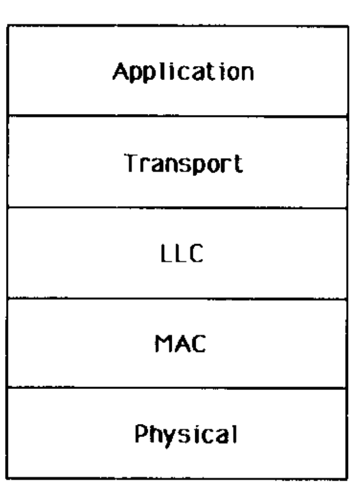


Fig. 2. LAN protocol structure.

2.1. Modelling of the token-passing MAC layer

The queueing model of the token-passing MAC layer is shown in Fig. 3. The token serves N MAC queues (stations and the bridge) in a cyclic manner. The service discipline is limited service to one [1]. Message arrive at the ith MAC queue according to the Poisson process at a rate λ_i, $i = 1, 2, \ldots, N$. Serve times of message at the ith MAC queue are independent and identically distributed (i.i.d.) random variables with first and second moments denoted by h_i and $h_i^{(2)}$, respectively. The station walking time, which is the time required to pass the token from station i to station $i+1$, is i.i.d random variable with mean r and variance δ^2, respectively. The utilization of the token at the ith queue is defined as:

$$\rho_i = \lambda_i h_i, \quad i = 1, 2, \ldots, N. \tag{1}$$

The total utilization of the LAN is then given by:

$$\rho = \sum_{i=1}^{N} \rho_i. \tag{2}$$

This paper deals only with the steady state of the system. It was shown by Kühn [3] that the following conditions are necessary and sufficient for stability of the token-passing system:

$$\rho < 1 \text{ and } \max(\lambda_i) Nr < 1 - \rho. \tag{3}$$

Boxma and Meister have given a very good approximate result for mean message waiting time for the above model. That is [2]

$$w_i = \frac{1-\rho+\rho_i}{1-\rho-\lambda_i r} \frac{1-\rho}{(1-\rho)\rho + \sum_{j=1}^{N} \rho_j^2} \times \left(\frac{\rho}{2(1-\rho)} \sum_{j=1}^{N} \lambda_i h_j^{(2)} + \frac{N\rho\delta^2}{2r} + \frac{r}{2(1-\rho)} \sum_{j=1}^{N} \rho_j (1+\rho_j) \right),$$
$$i = 1, 2, \ldots, N, \tag{4}$$

where w_i is the mean message waiting time at the ith queue. Then the total average message delay (the sum of the message waiting time, the mes-

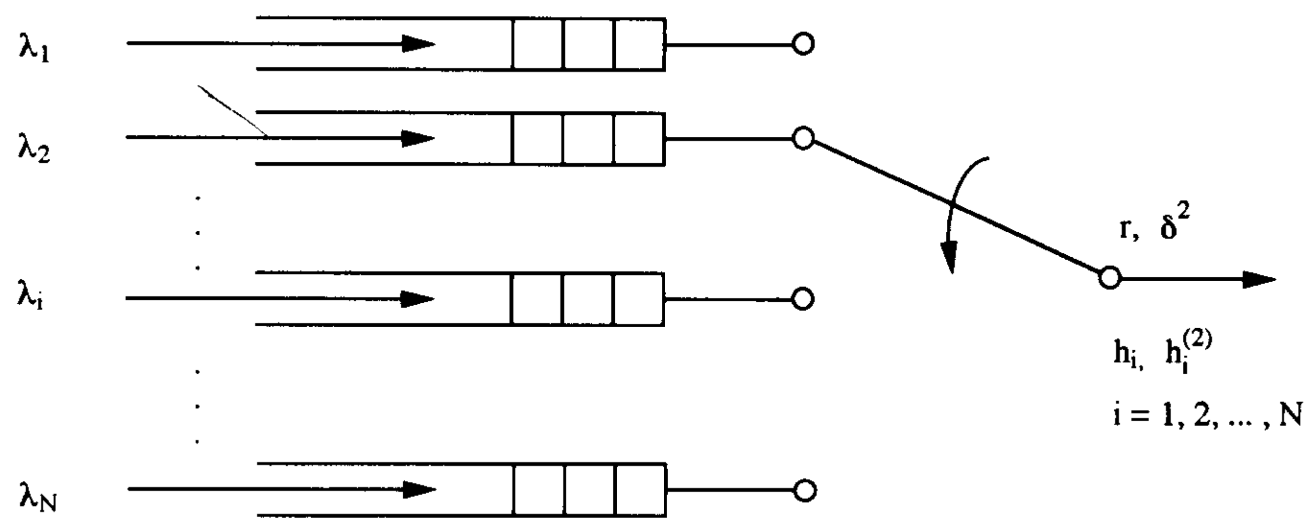


Fig. 3. Queueing model of the token-passing MAC layer.

sage transmission time and the propagation time between sender i and a randomly selected receiver) at the ith MAC-queue is given by:

$$f_i = w_i + h_i + Nr/2, \quad i = 1, 2, \ldots, N. \tag{5}$$

2.2. Modelling of end-to-end connections

We assume that the connection-oriented transport protocol operates a window flow control protocol to control message flow over virtual channels/ connections among designated stations. We follow the approach of [6,7] in modelling end-to-end connections. There are unidirectional connections between designated pairs of stations. Each connection has a source and a sink, and messages on a connection are individually acknowledged via message piggyback mechanism. The maximum number of messages on a connection is the window size of the connection.

To facilitate the following discussions, we classify stations into three types (see Fig. 4): single-stations, server-stations, and client-stations. A single-station can only communicate with another single-station over the same LAN, and single-station can support at most one connection. A station which communicates with a server is called a client-station. A server-station can support multiple connections with one connection to each client-station. A server-station receives service requests from client-stations, processes them according to FCFS discipline, and returns responses to the requested client-stations.

The above assumptions lead to the end-to-end queueing model of the internetwork shown in Figs. 5(a) and 5(b). Figure 5(a) shows the sub-

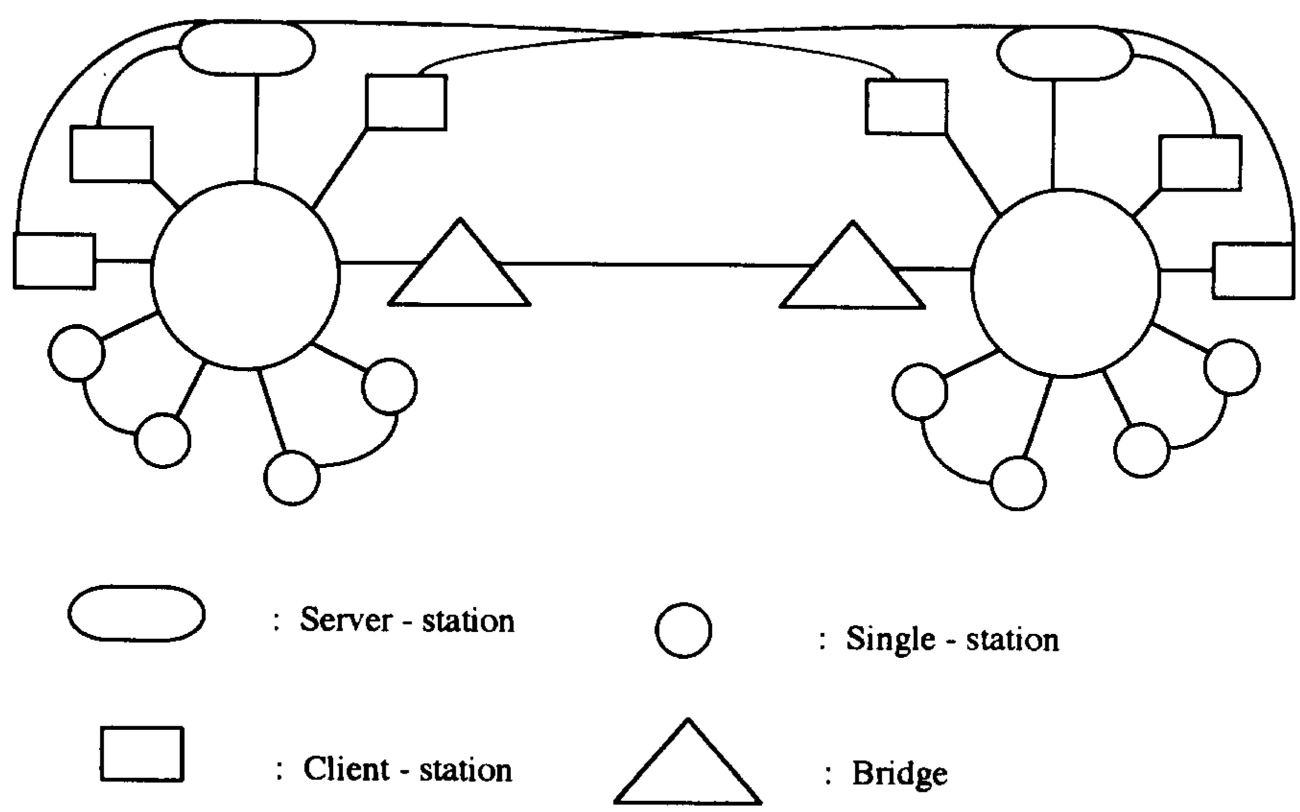


Fig. 4. Internetwork used for numerical examples.

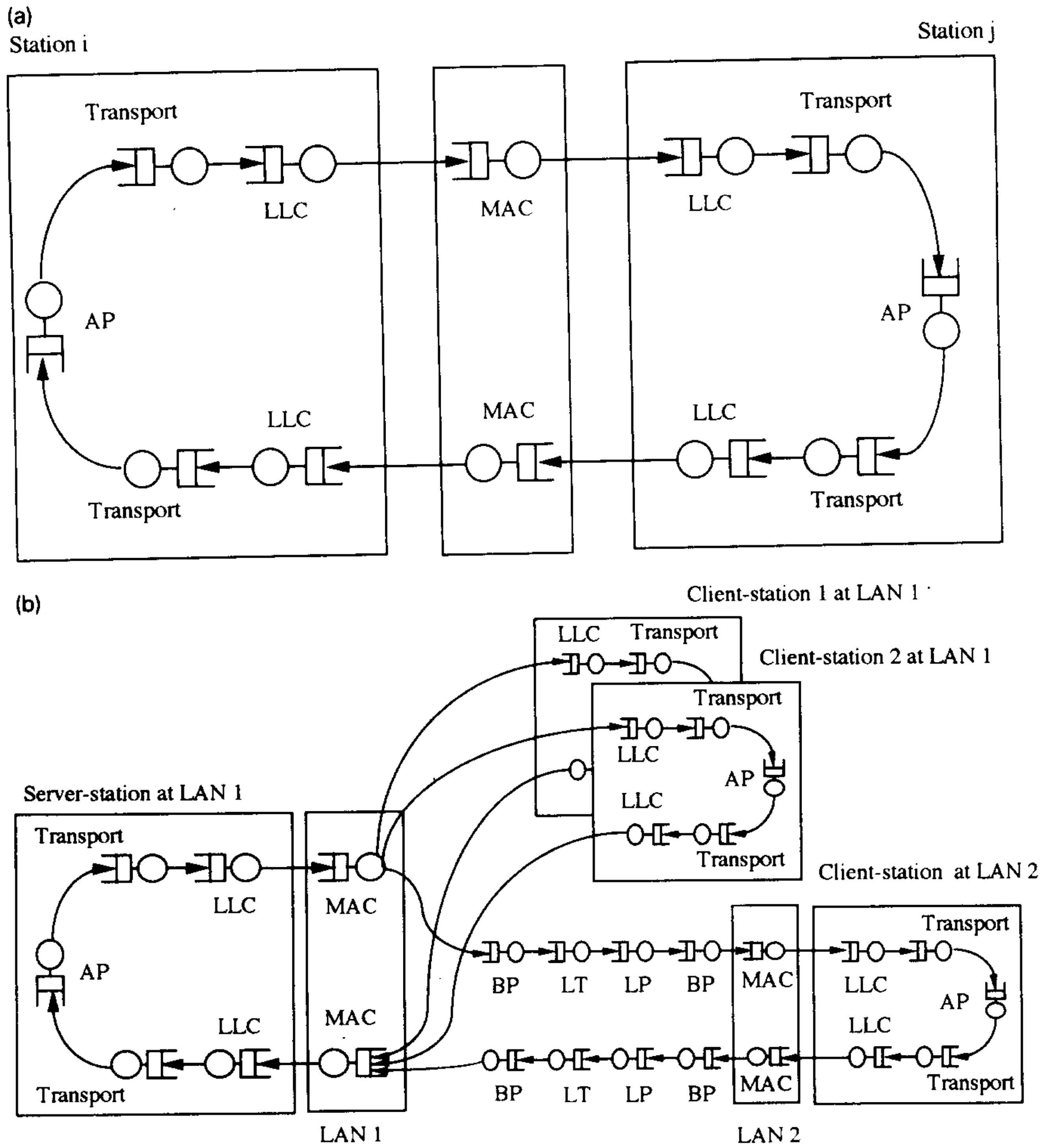


Fig. 5. End-to-end queueing model in the case of (a) a single-chain submodel, and (b) a multiple-chain submodel.

model for a connection between a pair of single-stations over the same LAN, which is a single-chain closed-queueing network. Figure 5(b) depicts the submodel for connections between a server-station and client-stations, which is a multiple-chain closed-queueing networks. There are two types of connections in this submodel, intra-LAN connection (corresponding to client-station located at the same LAN as the server-station) and inter-LAN connection (corresponding to client-station located at the remote LAN). The following features and assumptions are included in our model:

- No message fragmentation and reassembly take place throughout the entire transmission process of a message.
- The service time of the source corresponds to the interarrival time of messages which an application program at the source station can generate, and a message is generated at the source only if an acknowledgement is received and processed by the source transport layer. This is to ensure that the number of messages and the acknowledgements on a chain be equal to its assigned window size. Since data message piggy-back acknowledgements are used, the service time at the sink station corresponds to the interarrival time of the messages generated by the application program at the sink station. We assume a FCFS queue for the application program queue in the server-station with service time AP-server, while the application program queues in the client-stations/single-stations are modeled by IS queues with service time AP-client/AP-single.
- The LLC layer and the transport layer are modelled by FCFS queues for all stations with

service times denoted as LLC-single, LLC-client, LLC-server, transport-single, transport-client and transport-server, respectively.

- The chains interact with each other in the MAC layer. The MAC layer is modelled as IS queues with service time given by eq. (5).
- The service time for all single-stations and client-stations at the same layer (except MAC layer) are the same.

Moreover, the inter-LAN connection model in Fig. 5(b) has the following additional features:

- All the inter-LAN communication chains pass the two bridges and share the same internetwork link. The internetwork link is full duplex capable of handling inter-LAN traffic from both directions.
- The bridging function at the bridge is modelled as a FCFS queue with service time BP representing the bridge processing time.
- The internetwork link is modelled as a FCFS queue with service time LT corresponding to the link transmission time. The propagation delay between the two bridges is modelled as IS with service time equal to LP.

3. Solution algorithm to the end-to-end performance model

In this section, we present an algorithm for solving the single-chain and multiple-chain closed queueing networks given in Figs. 5(a) and 5(b) based on the MVA method.

3.1. MVA algorithm for the single-chain queueing network submodel

Define for queue j in the single-chain network of Fig. 5(a) the following equilibrium quantities:

- W single-chain population, i.e. the window size of the chain,
- Q a set of queues in the closed chain,
- τ_j mean service time,
- $n_j(W)$ mean queue length including message in service,
- $t_j(W)$ mean queueing time including message service time,
- $\lambda(W)$ throughput of the single-chain.

Then from Reiser [7] it follows that:

$$t_j(W) = \begin{cases} f_i, \\ \quad j = \text{MAC queue at single-station } i, \\ \tau_j, \\ \quad j = \text{AP queue at single-stations}, \\ \tau_j\left[1 + n_j(W-1)\right], \\ \quad j = \text{Transport or LLC queue} \\ \quad \text{at single-stations}, \end{cases} \tag{6}$$

$$\lambda(W) = \frac{W}{\sum_{k \in Q} t_k(W)}, \tag{7}$$

and

$$n_j(W) = \lambda(W) t_j(W). \tag{8}$$

Note that unlike the case in Murata and Takagi's algorithm, eqs. (6) to (8) cannot be solved alone based on MAC layer submodel given by eq. (5), because of the unknown arrival rate at the MAC queue for the bridge which can only be obtained in association with the MVA algorithm for the multiple-chain queueing network submodel described below.

3.2. MVA algorithm for the multiple-chain queueing network submodel

We now extend our solution method to the multiple-chain queueing network of Fig. 5(b). For the network we define the following notations:

- R number of chains in the closed network submodel (i.e., number of inter-chains and intra-chains);
- W^r window size of chain r, $r = 1, 2, \ldots, R$;
- $\mathbf{W}$ window size vector, which is $(W^1, \ldots, W^R)$;
- $R(j)$ set of chains visiting queue j, $j \in$ closed network submodel;
- $Q(r)$ set of queues in chain r, $r = 1, 2, \ldots, R$;
- τ_j^r mean service time of a chain r message at queue j, $j \in$ closed network submodel, $r \in R(j)$;
- $n_j^r(\mathbf{W})$ mean number of chain r messages waiting and being served at queue j, $j \in$ closed network submodel, $r \in R(j)$;
- $\lambda^r(\mathbf{W})$ throughput of chain r, $r = 1, 2, \ldots, R$;
- $t_j^r(\mathbf{W})$ mean queueing time of chain r messages at queue j, $r \in R(j)$, $j \in$ closed network submodel.

Then it follows from Reiser [7] that;

$$t_j^r(W)=\begin{cases} f_i, & i \text{ for given } r,\ j=\text{MAC queue},\\ \tau_j^r, & j=\text{AP queue of client-stations or LP queue},\\ \tau_j^r\left[1+n_j^r(W-e_r)\right], & j=\text{transport or LLC queue of client-station},\\ \tau_j^r\left[1+\sum_{k\in R(j)} n_j^k(W-e_r)\right], & j=\text{BP or LT queue, or AP or transport or LLC queue of server-station}, \end{cases} \tag{9}$$

$$\lambda^r(W)=\frac{W^r}{\sum_{j\in Q(r)} t_j^r(W)}, \tag{10}$$

and

$$n_j^r(W)=\lambda^r(W)t_j^r(W), \tag{11}$$

where

$$W-e_r \triangleq (W^1,\ldots,W^{r-1},W^r-1,W^{r+1},\ldots,W^R) \tag{12}$$

3.3. Solution algorithm for the performance model

Due to the interactions of chains at the MAC layer in both submodels, the above two MVA algorithms must be solved simultaneously to evaluate the end-to-end performance of the interconnected network. In the above MVA algorithms, f_i's obtained from eq. (5) are used as the service times for the MAC queues. In turn, the λ's obtained from the MVA algorithms can be used as the input values in eq. (5) for the MAC queues of single-stations and client-stations. In eq. (5), the arrival rates to the bridge/server-station MAC queues and the first and second moments of the message service times at the bridge/server-stations are respectively given by:

$$\lambda=\sum_{r\in R'}\lambda^r(W), \tag{13}$$

$$h=\frac{1}{\lambda}\sum_{r\in R'}\lambda^r(W)h_r, \tag{14}$$

$$h^{(2)}=\frac{1}{\lambda}\sum_{r\in R'}\lambda^r(W)h_r^{(2)}, \tag{15}$$

where R' is the set of chains in closed network visiting bridge/server-station MAC queue and h_r, $h_r^{(2)}$ are the first and second moments of service time at MAC-queue of chain r. The above relations lead us to an iterative solution algorithm similar to the one proposed in [6] for the end-to-end performance model:

(1) Set arrival rates λ_i, $i=1,2,\ldots,N$, to the MAC layer submodel to some initial values.
(2) Check λ_i's with the stability condition (3). If condition (3) is not satisfied, then modify the arrival rates small enough to meet the stability condition and calculate f_i using eq. (5).
(3) Calculate the throughput for each chain of Figs. 5(a) and 5(b) using the corresponding MVA algorithms. These values will be used in the next iteration cycle as arrival rates of each MAC queue according to eqs. (13)–(15). A proper modification of these input values may be needed to guarantee the convergence of the iterative process (see Appendix).

The convergence criterion for the iterations is defined by

$$\Delta_n=\sum_{i=1}^{N}\left|\lambda_i^{(n)}-\lambda_i^{(n-1)}\right|<\epsilon\ (\text{e.g.},\ \epsilon=10^{-6}), \tag{16}$$

for the nth iteration.

4. Numerical results and discussion

In this section, we present numerical results to demonstrate the end-to-end performance effects of system parameters (e.g. window size, server processing power and internet link capacity).

4.1. System configuration and parameters

In all the numerical examples we consider the example internetwork depicted in Fig. 4, where two identical token-rings are interconnected using bridges via the internetwork link. There are eight stations attached to each LAN. Among these stations, there are four single-stations, one server-station, two client-stations with intra-LAN connection to the server on the same LAN and one client-station with inter-LAN connection to the server at remote LAN.

The following parameters are used in our numerical examples:

- The LAN speed is kept constant at 4 Mbit/s. Message size follows the exponential distribution with an average length of 500 bytes. Then the message transmission time is also exponentially distributed with mean of $h = 1$ ms.
- The station walking time is exponentially distributed with mean $r = 0.005$ ms.
- Processing times per message at each layer of the network depend on the implementation of the protocols, the processor speed, buffer passing method, etc. The LLC processing time is assumed to be 1 ms for all stations. The processing times at the transport layer and AP layer in single/client-stations are 6 ms and 25 ms, respectively. High performance stations are assumed for the servers with processing time of 3 ms or 6 ms at

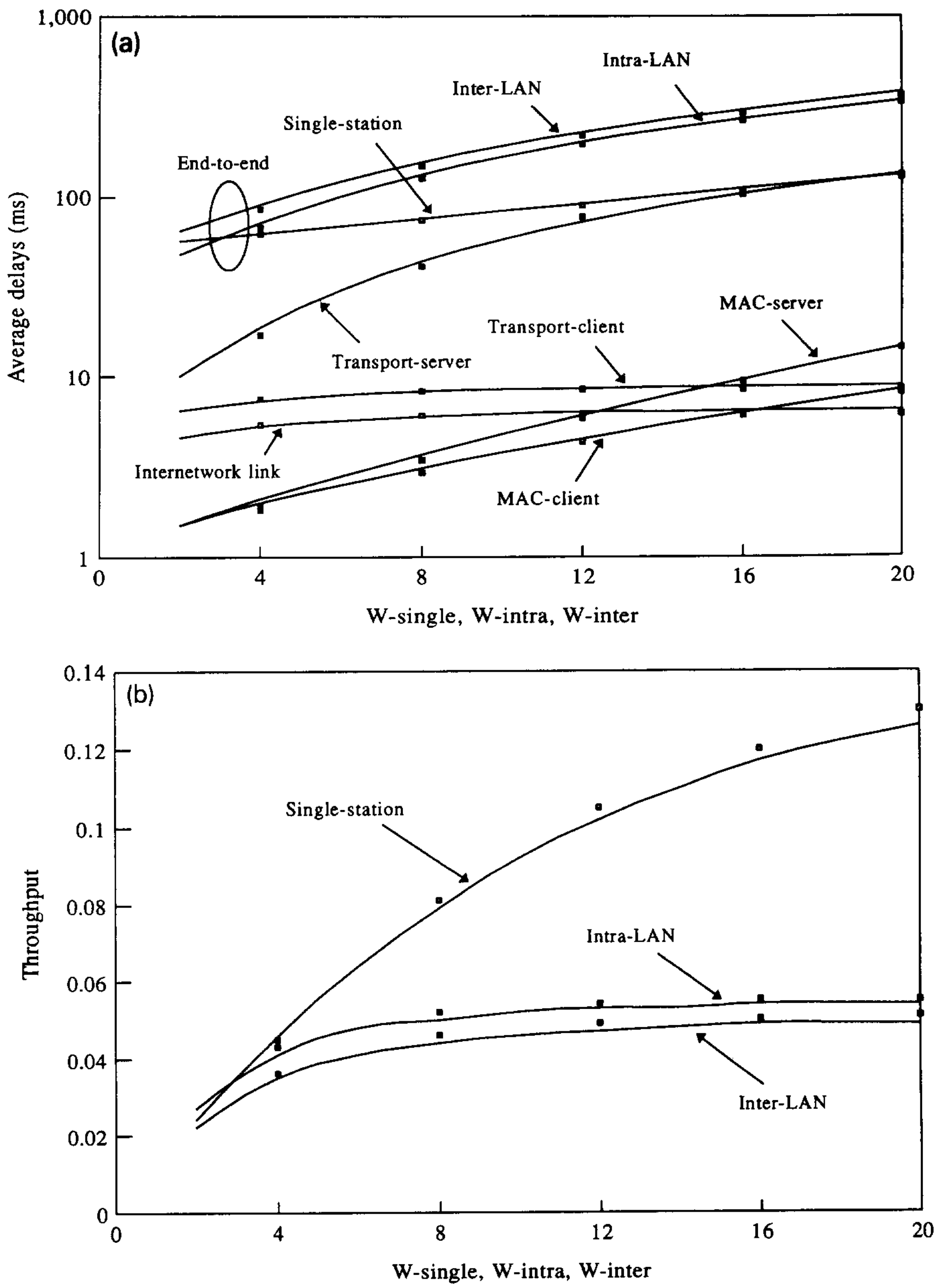


Fig. 6. (a) Average delays with AP-server = 5 ms, transport server = 6 ms, LT = 4 ms; (b) Throughput with AP-server = 5 ms, transport-server = 6 ms, LT = 4 ms.

the transport layer and 3 ms or 5 ms at the AP layer, respectively.

- The bridge processing time BP is assumed to be comparable with LLC processing time, and is 1 ms.
- The internet link speed can be 1 Mbit/s, 0.67 Mbit/s, or 0.33 Mbit/s, which corresponds to mean message transmission time LT = 4 ms, 6 ms and 12 ms, respectively, for an average message size of 500 bytes.
- Propagation delay LP between the two bridges is 0.05 ms.

4.2. Numerical and simulation results

Here, numerical results are presented and compared to the simulation results in order to validate the accuracy of the modelling approach. Computer simulation has been carried out based on *smpl* simulation program [5]. Throughout the

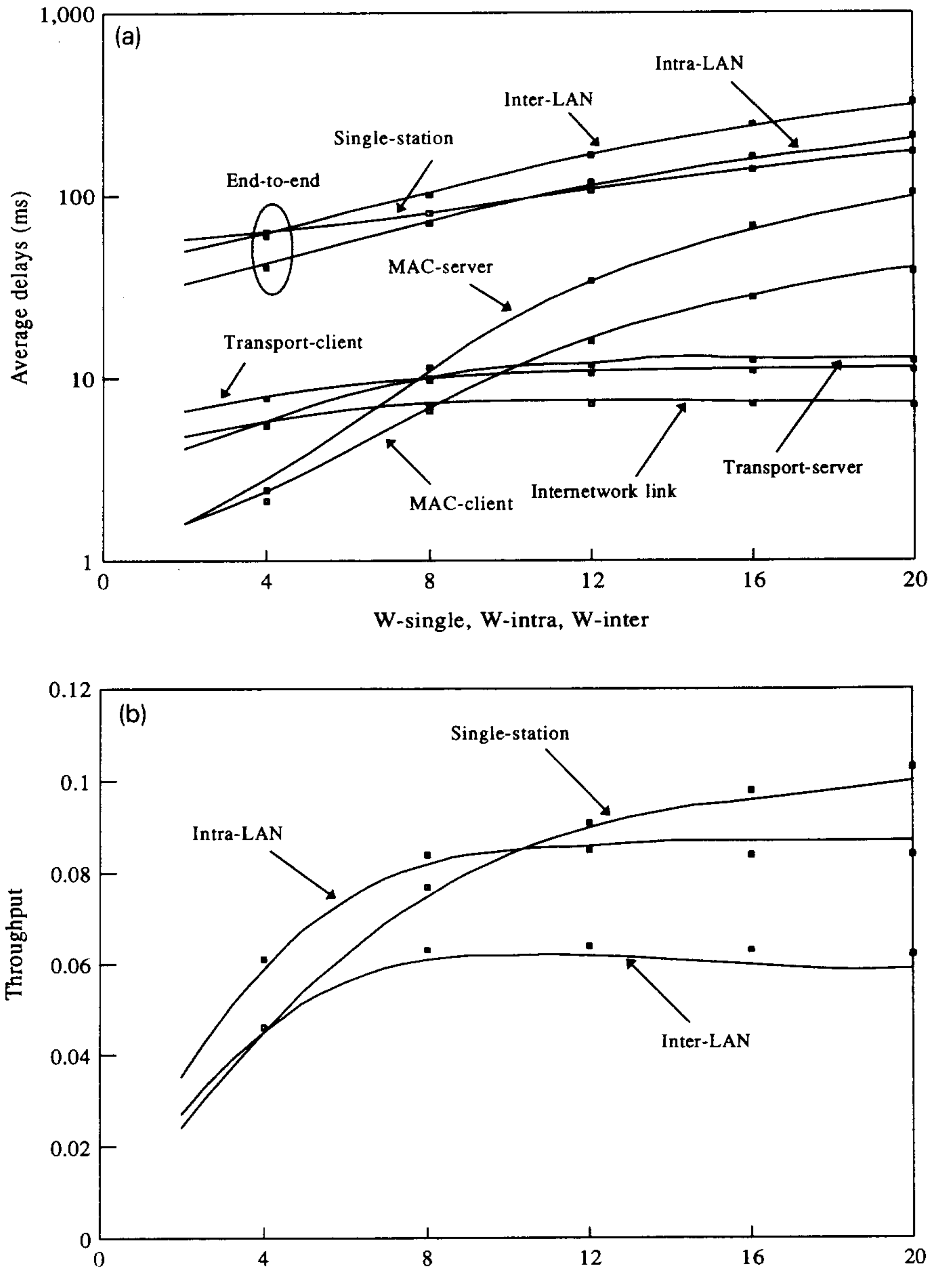


Fig. 7. (a) Average delays with AP-server = 3 ms, transport server = 3 ms, LT = 4 ms; (b) Throughput with AP-server = 3 ms, transport-server = 3 ms, LT = 4 ms.

following examples, the simulation results are shown as '■' and the numerical results are plotted by curves. We use W-single, W-intra, and W-inter to denote the window sizes of the single-station connection, intra-LAN connection, and inter-LAN connection, respectively. Figures 6(a) and 6(b) show the internetwork performance for the case of W-single = W-intra = W-inter = 1, 2, ..., 20, LT = 4 ms, transport-server = 6 ms, AP-server = 5 ms. Figure 6(a) shows the average delays at MAC queues, transport queues and the internetwork link queues. Average end-to-end delays for single-LAN, intra-LAN and inter-LAN connections are also shown in the figure. The throughput per single-station, intra-LAN and inter-LAN connection are plotted in Fig. 6(b). These figures indicate that numerical results are in good agreement with the simulation results.

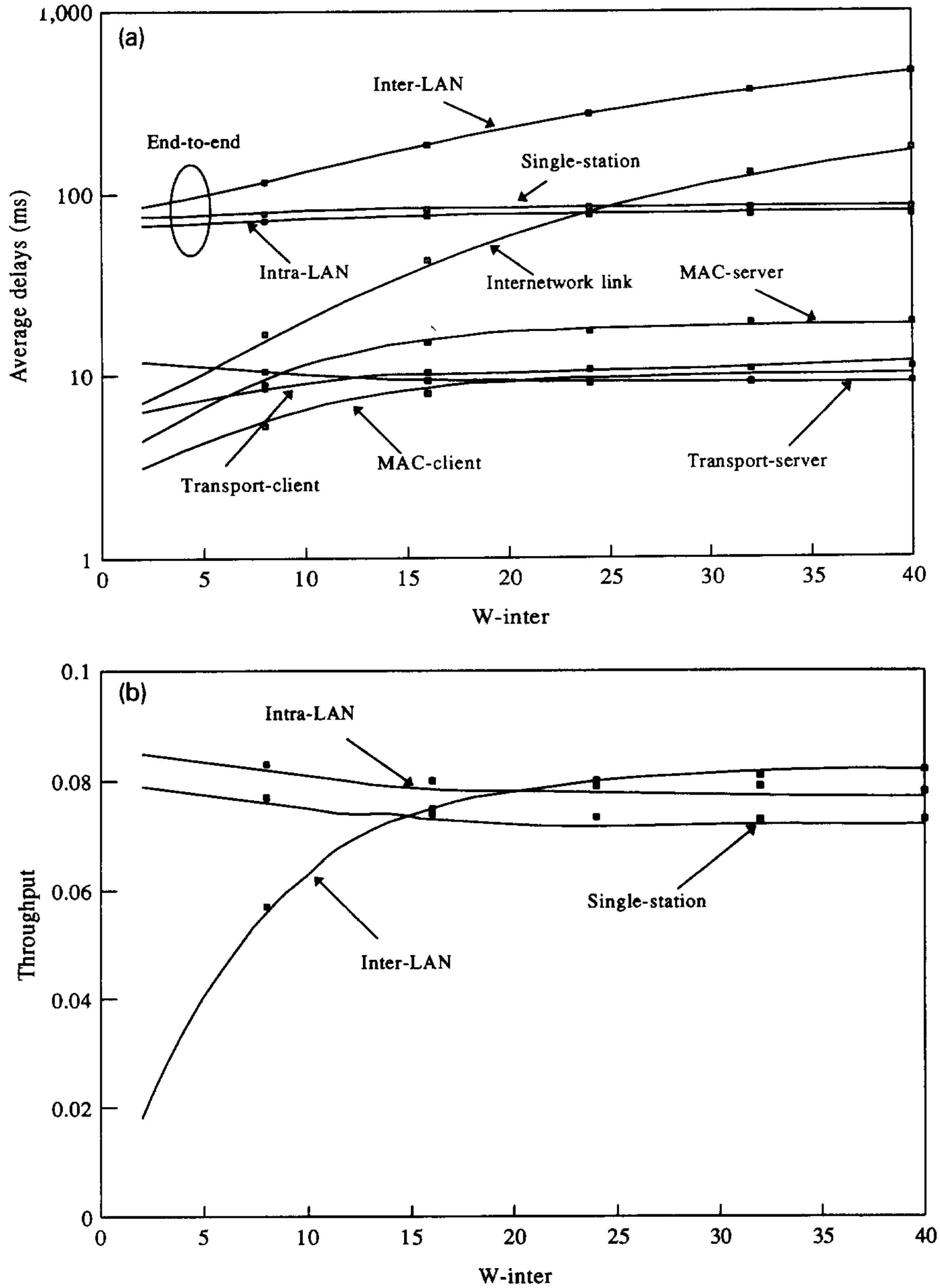


Fig. 8. (a) Average delays with W-single = W-intra = 8, AP-server = transport server = 3 ms, LT = 6 ms; (b) Throughput with W-single = W-intra = 8, AP-server = transport-server = 3 ms, LT = 6 ms.

From Fig. 6(a) we see that delays at MAC layer increase monotonically as the window size goes up. The bottlenecks reside at transport layer of server-stations. Figure 6(b) shows that the throughput for single-station connection is much larger than that of intra-LAN and inter-LAN connections. This is because the delays at transport layer of the single-stations are much smaller than those at the server-stations.

Next, we demonstrate the effect of changing the processing time of server-stations at transport layer and AP layer. Figures 7(a) and 7(b) show the average delays and throughput for the case with W-single = W-intra = W-inter = 1, 2, ..., 20, LT = 4 ms, transport-server = AP-server = 3 ms. In this case, the system bottlenecks still reside at the transport layer for small window sizes; however, as the window sizes reach such values that

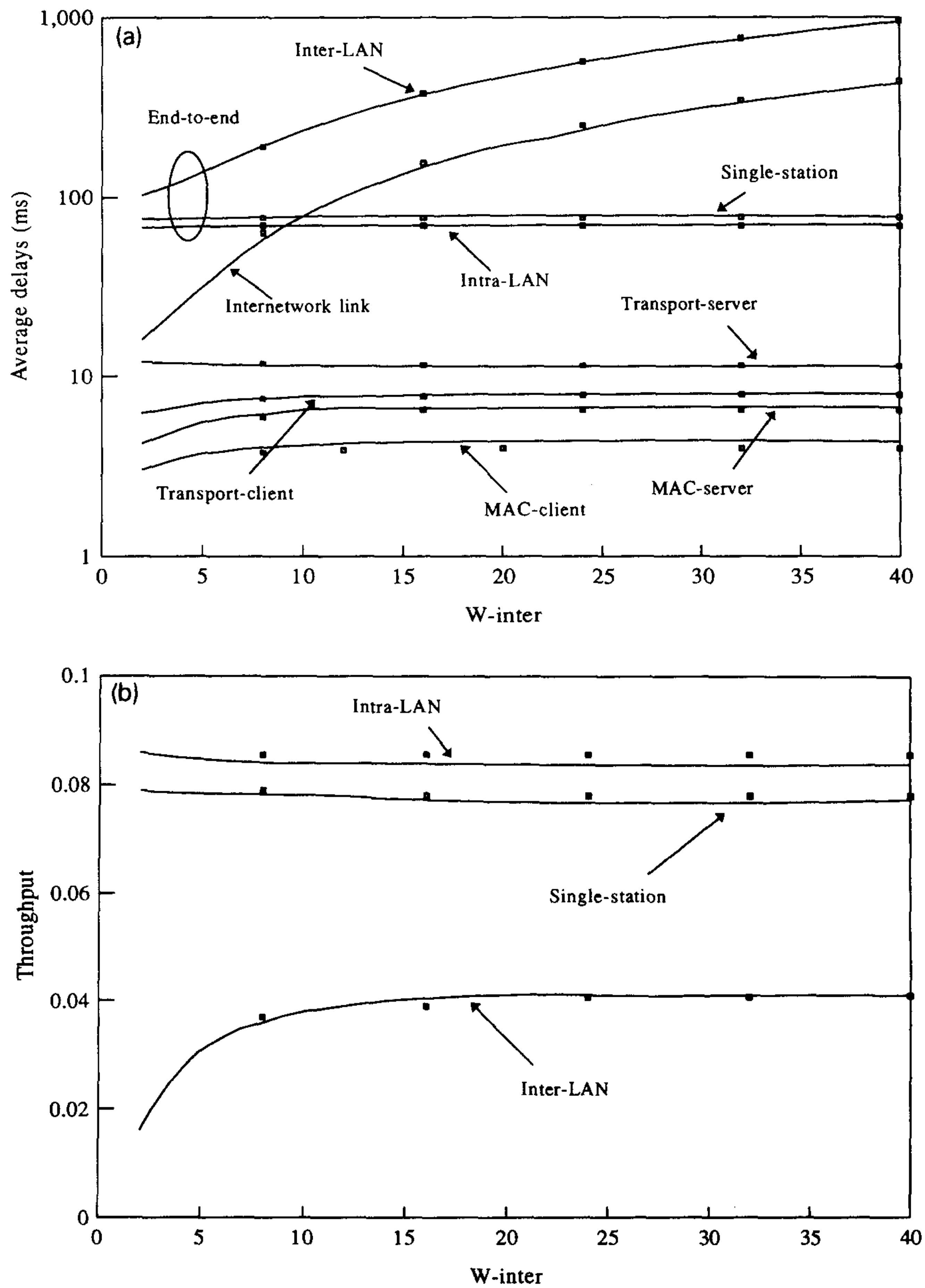


Fig. 9. (a) Average delays with W-single = W-intra = 8, AP-server = transport server = 3 ms, LT = 12 ms; (b) Throughput with W-single = W-intra = 8, AP-server = transport-server = 3 ms, LT = 12 ms.

the system throughput becomes saturated (this happens in Fig. 7(b) for window size larger than 10), the bottlenecks shift to the MAC layer. From Fig. 7(b) we see that throughput of intra-LAN and inter-LAN connection increase significantly compared with the low speed server-station case.

The influence of the internetwork link transmission capacity on system performance is presented in Figs. 8 and 9, where we compare the average delays and throughput with variable transmission capacity of the internetwork link for LT = 6 ms and 12 ms, and W-single = W-intra = 8, W-inter = 1, 2, ..., 40, Transport-server = AP-server = 3 ms. From Figs. 8(a) and 9(a) we can see that, except for very small values of W-inter, the internetwork link is the bottleneck. This is true in almost all practical systems where the internetwork link capacity is smaller than the transmission capacity of the LANs. We observe from Fig. 8(b) that when W-single and W-intra are fixed, the throughput per inter-LAN connection can be varied by changing the window size W-inter, and it is possible for inter-LAN throughput to exceed the intra-LAN throughput (per connection), even with the internetwork link as the bottleneck. However, as we can see from Fig. 9(b) that this situation will change when the transmission capacity of the internetwork link is reduced to a certain level. In this case, due to the increased link processing time, the inter-LAN throughput per connection is always less than the intra-LAN throughput per connection, no matter what values W-inter may take.

The impact of the bridge processing time to the internetwork performance is similar to that of the internetwork link transmission capacity. Therefore, the corresponding curves are not included in this paper.

5. Conclusion

In this paper we have investigated the end-to-end performance of interconnected LANs with server/client configuration. The end-to-end performance model of the interconnected network was built based on single-chain and multiple-chain closed queueing network models. An iterative algorithm for multiple-chain closed-queueing network was developed based on the MVA method. Numerical examples and computer simulations were presented to show the performance characteristics of the interconnected system. Our results indicated that the window size, server processing speed and internetwork transmission capacity have significant influences on the end-to-end interconnected network performance.

Appendix

Solution algorithm for a multiple-chain closed-queueing network

First, let us start with the algorithm proposed by Murata and Takagi [6] for a single-chain closed-queueing network and observe what happens when it is applied to multiple-chain closed-queueing network. The algorithm for a single-chain closed-queueing network is stated as follows [6]:

(1) Initialize arrival rates $\lambda_i^{(0)}$ and λ_i^{base} for all i.

(2) (Iteration cycle) check $\lambda_i^{(n)}$'s with the stability condition (3). If condition (3) is not satisfied, then

$$\lambda_i^{(n)} \Leftarrow \frac{\lambda_i^{\text{base}} + \lambda_i^{(n)}}{2} \tag{A.1}$$

(3) Calculate the mean message waiting times f_i using (5).

(4) For all closed-queueing networks do:

(a) solve for the throughput of each chain using (6) to (8) and denote it $\lambda_i'^{(n)}$, $\forall i$.

(b) If the derived throughput $\lambda_i'^{(n)}$ is greater than $\lambda_i^{(n)}$, then let

$$\lambda_i^{\text{base}} \Leftarrow \lambda_i^{(n)}, \tag{A.2}$$

$$\lambda_i^{(n)} \Leftarrow \frac{\lambda_i'^{(n)} + \lambda_i^{(n)}}{2}. \tag{A.3}$$

Otherwise, let

$$\lambda_i^{(n)} \Leftarrow \frac{\lambda_i^{\text{base}} + \lambda_i^{(n)}}{2}. \tag{A.4}$$

(5) Let $n = n + 1$, go to step (2) for continuing iteration until the criterion is met.

In the above algorithm, λ_i^{base} is the maximum value which guarantees the iteration does not diverge in the current step. Obviously, the iteration cycle will converge to λ_i^{base} if the derived throughput $\lambda_i'^{(n)}$ is kept smaller than the pervious input, $\lambda_i^{(n)}$. However, in the case of multiple-chain closed-queueing network, the throughputs for different chains interact with each other at the MAC layer so that the changes in mean message waiting times for chains are not consistent, which may lead to the derived output $\lambda_i'^{(n)}$ converge to some other value when the input $\lambda_i^{(n)}$ converges to λ_i^{base}. Therefore, the modification of λ_i^{base} becomes necessary when $\lambda_i'^{(n)}$ does not converge to λ_i^{base}. Our iterative algorithm for a multiple-chain closed-queueing network is then given as follows based on the above observation:

(1) Initialize arrival rates $\lambda_i^{(0)}$ and λ_i^{base} for all i.
(2) Follow the steps 2) to 4) as mentioned above.
(3) If

$$\Delta'_n = \sum_{i=1}^{N} \left| \lambda_i'^{(n)} - \lambda_i'^{(n-1)} \right| < \epsilon' \quad (\text{e.g., } \epsilon' = 10^{-8}), \tag{A.5}$$

then let

$$\lambda_{\mathrm{i}}^{\text{base}} \Leftarrow \frac{\lambda_{\mathrm{i}}'^{(\mathrm{n})} + \lambda_{\mathrm{i}}^{base}}{2}, \tag{A.6}$$

and

$$\lambda_i^{(n)} \Leftarrow \frac{\lambda_i^{(n)} + \lambda_i^{\text{base}}}{2}. \tag{A.7}$$

(4) Let $n = N + 1$, go to step (2).

The numerical examples showed that the iterations which do not converge by using the algorithm proposed by [6] will reach the right points by using our modified algorithm. It also showed that the iteration times depends on the system parameters, such as window size and processing time. For example, we needed 5 iterations to converge for the example presented in Fig. 6 with W-single = W-intra = W-inter = 2 while 32 iterations for W-single = W-intra = W-inter = 20. When transport-server = AP-server = 3 ms as in the case of Fig. 7, 1292 iterations were required for W-single = W-intra = W-inter = 20.

References

[1] ANSI/IEEE Standard 802.5, *Token Ring Access Method* (IEEE Press, New York, NY, 1985).

[2] O.J. Boxma and B. Meister, Waiting time approximations for cyclic-service systems with switchover times, *Performance Evaluation* **7** (1987) 299–308.

[3] P.J. Kühn, Multiqueue system with non-exhaustive cyclic service, *Bell System Tech. J.* **58** (1979) 671–698.

[4] K. Kummerle, J.O. Limb and F.A. Tabagi, *Advances in Local Area Networks* (IEEE Press, New York, 1978).

[5] M.H. MacDougall, *Simulating Computer Systems* (MIT Press, Cambridge, MA, 1987).

[6] M. Murata and H. Takagi, Two-layer modelling for local area networks, *IEEE Trans. Comm.* **36** (9) (1988) 1022–1033.

[7] M. Reiser, A queueing network analysis of computer communication networks with window flow control, *IEEE Trans. Comm.* **27** (8) (1979) 1199–1209.