Performance of Buffered Multistage Interconnection Networks in a Non-Uniform Traffic Environment[1]

M. Atiquzzaman
Dept. of Electrical & Computer Systems Engineering
Monash University, Clayton 3168, Melbourne, Australia.
Tel: +61 3 905 5383, Fax +61 3 905 3454
atiq@eng.monash.edu.au

M.S. Akhtar
Dept. of Electrical Engineering
King Fahd Univ. of Petroleum & Minerals
Dhahran 31261, Saudi Arabia.

[1] A preliminary version of this paper appears in 7th IEEE International Parallel Processing Symposium, April 1993, Newport Beach, California.

Proposed Running Head: Buffered Multistage Networks in Non-Uniform Traffic.

Address for sending proofs:

M. Atiquzzaman
Dept. of Electrical & Computer Systems Engineering
Monash University, Clayton 3168, Melbourne, Australia.
Tel: +61 3 905 5383, Fax +61 3 905 3454

Abstract: Multistage interconnection networks (MIN) are used to connect processors to memories in shared memory multiprocessor systems. They have also been proposed as switching fabrics in ATM switches in Broadband ISDN networks. A multistage network consists of several stages of switching elements (SE) connected by a permutation function. Since routing conflicts among packets at the SEs cause blocking and reduce the performance of the network, analytical models have been developed and simulation studies carried out for performance evaluation of MINs in the presence of uniform traffic in the network. Buffers are used at SEs to increase the performance of the network.

The suitability of two existing models for possible use in the presence of hot spot is investigated in this paper. Results show that errors from the models are large. In this paper, we propose a new and efficient Markov chain model for performance evaluation of a single buffered Omega network in the presence of hot spot. Results obtained from the existing models and the proposed model for the uniform and the hot spot traffic patterns are presented. It is shown that the proposed model is better than existing models in the presence of hot spot traffic. While the existing models produce small errors in the case of low blocking in the network, the errors become prominent with increased blocking. The proposed model produces accurate results in the more general case of hot spot traffic, whereas the existing models produce accurate results only in the special case of uniform traffic.

Index terms: Multistage interconnection networks, buffered networks, performance evaluation, analytical modeling, Markov chain, hot spot traffic, network bandwidth.

List of symbols:

τ – Tau

α – Alpha

ρ – Rho

β – Beta

1 Introduction

An interconnection network is used to connect processors to memories in a multiprocessor system. Different types of networks have been proposed in the literature. Because of cost effectiveness, multistage interconnection networks (MIN) are suitable for interconnecting a large number of processors to memories in multiprocessor systems [5, 17]. MINs have also been proposed as switching fabrics for packet switching in ATM switches of Broadband ISDN networks.

A MIN consists of several stages of small switching elements interconnected using a permutation function. Routing conflicts among packets at the SEs result in some packets being blocked or discarded. A MIN is called a buffered MIN if the SEs contain internal buffers. Buffered networks prevent internal loss of data in case of routing conflicts. Under uniform traffic conditions buffered MINs have significantly higher throughput than unbuffered ones [4]. In the presence of non-uniform traffic, an unbuffered network can have a higher throughput than a buffered one [1].

Since the performance of a MIN plays a critical role in the overall performance of a multiprocessor system, both unbuffered and buffered networks have been widely studied using analytical models and simulation techniques. Most of the analytical models have assumed uniform traffic in the network. Under uniform traffic conditions, the probability of a packet entering the network to be destined for any output link is the same for all output links. Performance of *unbuffered* MINs under uniform traffic conditions have been reported in [15, 10, 3]. Patel [15] has developed a recursive technique, while Bhuyan [3] has developed an analytical model for bandwidth evaluation of an unbuffered Omega network where each processor has a unique favorite memory. The performance of unbuffered Banyan networks has been analyzed by Kruskal [10]. Performance in the presence of hot spot traffic pattern has been reported in [1, 2, 16]. Favorite output has been discussed in [3, 11].

Performance of *buffered* MINs in uniform traffic has also been reported in the literature. Dias [4] used the timed Petri net to show that the performance of a buffered network is comparable to that of a crossbar. A discrete Markov chain model to evaluate bandwidth was described by Thanawastien [20]. He defined the states of the chain by the number of packets in the buffers of a stage. Jenq [7] proposed an elegant iterative Markov chain solution for a MIN using 2×2 SEs by splitting the stage cycle into two cycles. His work was generalized to $a \times a$ SEs by Yoon [24]. Theimer [21] removed two independence assumptions of Jenq to obtain better results. Based on Jenq's technique of analysis, Hsiao [6] proposed a model by relaxing an independence assumption of Yoon. Performance of *buffered* MINs under different types of non-uniform traffic environment has been reported in [11, 8, 9, 14].

Pfister [18] discussed the phenomenon of buffer saturation arising as a result of a hot spot in a buffered MIN. In the case of a hot spot in a network, a packet entering the network has a higher probability of being directed to a particular output link when compared to other output links of the network. A hot spot arises in a shared memory multiprocessor system due to the use of shared variables used for locking, barrier synchronization, pointers to shared queues etc. These are indivisible primitives and must be stored in a single shared memory which becomes a hot memory. The other memory modules will be called cold memories. The primitives are accessed by all processors, giving rise to an increased request rate for the hot memory module. The phenomenon is called *hot spot contention* and gives rise to *tree saturation* in buffered networks. A tree of saturated buffers, rooted at the hot memory module and having leaves at the processors, obstructs the flow of hot traffic, as well as any cold traffic that has to cross the saturated tree. Tree saturation results in degradation in the performance of the network. Combining and feedback schemes have been suggested as partial solutions to the problem [13, 19, 23]. The degradation of an unbuffered network in the presence of a hot spot is due to the high rate of contention at SEs carrying the hot traffic.

Performance of MINs in the presence of hot spot traffic is usually obtained from simulation studies. Unfortunately, such simulation studies are very expensive in terms of computation time. Analytical models for performance evaluation of MINs under uniform traffic conditions are available in the literature but no efficient analytical models are available to study the performance of MINs in the presence of hot spot traffic. Analytical models however, permit a fast and inexpensive method of performance evaluation.

The aim of this paper is to develop an analytical model for performance evaluation of single-buffered MINs in the presence of a hot spot. We modified two existing models to assess their accuracy in the presence of hot spot traffic. Since the models were not accurate, we investigated the cause of errors in these existing models. We found that the models did not take a complete account of blocking in the SEs, and therefore gave poor results in the case of significant blocking in the network. In this paper, we propose a new Markov chain model which provides very accurate results for both uniform and hot spot traffic patterns. The greater accuracy of the proposed model is due to taking a rigorous account of blocking in the SEs.

An exact analysis of a buffered MIN using Markov chains is not practical even in the uniform traffic environment, because it leads to a multidimensional state space with one dimension for every buffer of the network. Since this huge state space cannot be handled with reasonable effort, simplifying assumptions regarding independence of buffers are made to reduce the size of the state space. Jenq [7] laid the groundwork for analyzing a single-buffered Omega network having 2×2 SEs. He made the following *independence assumptions* in order to reduce the state space of the model.

1. The state of an SE at stage k is statistically not distinguishable from that of another SE at the same stage.

2. Routing requests from packets in a buffer at successive stage cycles are independent.

3. The two buffers in the same SE are statistically independent.

Based on the above assumptions, Jenq proposed a 2-state model where the network was modeled by the state of a single buffer being empty or full. Jenq's assumptions do not hold in case of blocking in the network as described below.

A packet blocked during a cycle for a particular output link of an SE will request the same output link of the SE at all successive cycles until it moves to the next stage.

Therefore, assumption 2 of Jenq's model does not hold for blocked packets. Secondly, the probability that a packet in a buffer of an SE can move forward during a cycle *depends* on the state of the other buffer of the SE during that cycle. For example, a packet in a buffer of an SE has a high probability of moving forward to the next stage if the other buffer of the SE is empty. The probability is low when the other buffer is in a blocked state. The above fact is in contradiction to assumption 3 of Jenq's model. Because of the above independence assumptions, the results obtained from the model have large errors even for the uniform traffic pattern.

To investigate the extent of error caused by the independence assumptions, Jenq proposed a second model by taking into account the dependencies between the buffers of an SE. This did not result in a significant improvement in the results over the first model. He therefore, concluded that *the assumption of independence between buffers of an SE is a good approximation.*

Jenq's work has been extended by Yoon [24] to model a multiple-buffered Delta network using $a \times a$ SEs. The same independence assumptions as used by Jenq have been used. Results obtained from the model also exhibit significant errors even for uniform traffic.

Theimer [21] developed a model by removing independence assumptions 2 and 3 of Jenq. To take into account the effect of blocking, the state of a buffer was described as being empty, containing a new packet, or containing a blocked packet. Theimer concluded that *consideration of the dependencies between buffers of an SE leads to a small improvement of the results.* Results obtained from the model are close to simulation results for *uniform traffic.*

Hsiao [6] proposed an extension to Yoon's model by taking into account the dependency between states of the buffers of an SE and the fact that a blocked packet will still go to a previously determined output link. The inaccuracy in the model is due to its inability to store the information regarding the output link of the SE for which the packet was blocked.

We modified the above mentioned existing models in the hope of arriving at a model which would be suitable for hot spot traffic. The assumptions for uniform traffic which do not hold in the case of hot spot traffic are as follows:

- The probabilities of a new packet at a hot SE to be routed to the output links of the SE are not equal in the case of hot spot traffic. The routing probabilities of a new packet to the output links are proportional to the sum of the probabilities of the requests to the set of memories that can be reached through the output links.

- Because of different traffic rates at the SEs of a stage in the presence of hot spot traffic, the SEs in a stage are not statistically identical. Stage i will have i types of SEs classified according to data rates at the inputs to the switches (see Section 2.3).

By incorporating the above modified assumptions, we found that the existing models produced large errors in the presence of hot spot traffic (see Section 4). We analyzed the models carefully to determine the cause of the errors, and found that they did not take a complete account of the fact that a blocked packet in subsequent cycles will request the same output link (of the SE) which it had requested when it got blocked. The shortcoming of the previous models is their inability to memorize the output link which was requested when a packet got blocked. Hence they *implicitly assume (in the analysis) that a blocked packet has equal probabilities of requesting the output links* of the SE during subsequent cycles. The impact of this implicit assumption is reflected in increased errors as blocking in the network increases. The assumption produces small errors in the case of uniform traffic, but the errors get highly magnified and produce clearly unacceptable results in the case of hot spot traffic. The increased errors in the case of hot spot traffic are due to increased blocking in the SEs because of tree saturation.

To account for the fact that a blocked packet in an SE will request the same output link of the SE at successive stage cycles, we propose a 4-state Markov model which memorizes information regarding the output link that was requested when a packet got blocked. In addition to removing both the independence assumptions of Jenq and Yoon, we have taken a rigorous account of the dependencies among successive requests from a buffer in an SE.

Jenq and Theimer have assumed that all the SEs at a particular stage are identical for uniform traffic in the network. In the case of hot spot traffic in the network, the traffic rates at the SEs at a stage are not necessarily the same (see Section 2.3). This results in different sets of equations for the SEs having different traffic rates – the basic model remains the same as for the uniform traffic case. For example, only one set of equations is required for the hot SEs throughout the network (except those at the last stage). A second set of equations is required for the cold SEs which have their inputs connected to the cold output links of the hot SEs at the previous stage. It can be shown that n different sets of equations are required for an n-stage network. We have extended the models proposed by Jenq and Theimer to make them suitable for hot spot traffic by having different sets of equations for the different types of SEs. In a similar manner, we have extended our proposed model to enable analysis in the presence of hot spot traffic.

Because of space limitations, our proposed model will be described only for the uniform traffic. Extensions to the model for hot spot traffic pattern can be done as mentioned in Section 5. However, *results obtained from Jenq's model, Theimer's model, and our proposed model will be presented for both uniform and hot spot traffic patterns.*

The objectives of the research work described in this paper were as follows.

- To modify the existing analytical models for hot spot traffic, and evaluate their accuracy.

- To develop a new generalized analytical model for performance evaluation of buffered MINs in the presence of hot spot traffic. Uniform traffic can also be analyzed by the generalized model since uniform traffic is a special case of hot spot traffic where the probability of hot memory request is the same as that of a cold memory.

- To check the validity of the proposed model by comparison with simulation results.

The Omega network (Figure 1) will be used as an example of a MIN to be modeled. The rest of the paper is organized as follows. The Omega network, the modeling assumptions, and properties of the network under hot spot traffic pattern are described in the next section. A new analytical model for performance evaluation of a single-buffered Omega network is proposed in Section 3. Results obtained in the presence of uniform and hot spot traffic from the proposed model are compared with those from previous models and stochastic simulations in Section 4, followed by concluding remarks in Section 5.

2 Omega Network and Modeling Assumptions

The Omega network [12] will be taken as an example of a MIN to be modeled. This section describes the Omega network, its properties under hot spot traffic, and the assumptions under which the analytical model is developed.

2.1 The Omega Network

The Omega network connects N processors to N memories using $n = \log_2 N$ stages of $N/2$ SEs per stage. The $i-$th stage will be denoted by S_i, $1 \leq i \leq n$. A perfect shuffle permutation is used to connect adjacent stages as shown in Figure 1. Each SE is a $a \times a$ crossbar allowing any input link to be connected to any one of the output links. Each input link of an SE has buffers to hold packets. A conflict arises if two packets at the two buffers of an SE request the same output link simultaneously. Arbitration schemes are used to resolve such conflicts.

A request generated from a processor is bundled into a packet consisting of data and the destination address. The destination address is an $n-$bit number represented by $D = (d_1 d_2 \ldots d_{n-1} d_n)_2$. *Destination tag* routing is used to route the packets through the network. An SE at S_i inspects bit d_i, and in case of no conflict routes the packet to the upper or lower output of the SE depending on d_i being 0 or 1 respectively. A *unique path* of *constant length* exists between any input-output pair of the network, thereby rendering the Omega network a blocking type of network.

2.2 Assumptions

For modeling purposes, the following assumptions are made regarding the network and its operation.

1. There are $N = 2^n$ processors and N memory modules in the system, where n is an integer. The $i-$th processor and the $j-$th memory will be denoted by PE_i and MM_j respectively, where $0 \leq i, j \leq N - 1$. Because of reduced complexity for smaller switch sizes, we assume 2×2 SEs.

2. *Synchronous operation* of the network is assumed. All processors request memories at the beginning of a network cycle, and requests arriving at memory modules are serviced at the end of a network cycle. The service times of all memory modules are also equal.

3. *Packet switching* is assumed for routing of messages. A packet contains data and the destination address.

4. *Fair routing* arbitration logic at the SEs is assumed. If two packets at the input buffers of an SE request the same output link, the SE randomly selects one input and the rejected one is blocked in the buffer and has to try again in the next cycle.

5. *Temporal independence* of requests is assumed, i.e. the request generated by a processor in a cycle is independent of whether requests at previous cycles were accepted or rejected by the network.

6. Requests are assumed to be *spatially independent*, i.e., requests generated by a processor are independent of requests generated by other processors.

7. Processors generate *random request* at the beginning of a memory cycle. The probability that a processor generates a memory request at the beginning of a cycle is p_0. Thus p_0 is the average number of requests generated per cycle by each processor.

8. For uniform traffic pattern, the probability that a processor request is directed for any particular memory module is $1/N$. For hot spot traffic pattern, MM_h will be assumed to be a *hot memory module* for all processors. If PE_i generates a request, the probability that it will request MM_h is h, whereas the probability of requesting

any module MM_j, $j \neq h$, is $h' = (1-h)/(N-1)$ where $h > h'$. The cold memory modules are equally likely to be referenced.

9. In case of uniform traffic, a new packet arriving at the input of an SE is equally likely to request any output link of the SE. On the contrary, for hot spot traffic pattern, a packet arriving at the input of an SE is not necessarily uniformly distributed over the outputs of the SE.

10. Each SE is assumed to have a single buffer at each of its input links. There is *no loss of packets* inside the network. A backpressure mechanism prevents packets from entering the network if it is unable to accept new packets due to full buffers. Input buffers have the advantage of simplifying the flow control signals (required for the backpressure mechanism) [11], and receive one packet per cycle thereby reducing the control circuitry of the SE.

11. There is *no blocking at the outputs of the network*. The output links of the network are at least as fast as the internal links.

The assumptions are the same as those used by most authors. In particular, all of the above assumptions (except the non-uniform traffic distribution) have been used in [7, 21]. We retain the assumptions to enable a fair comparison between the previous models and the proposed model. Assumption 5 makes the analysis simple without introducing significant errors as shown in [15]. An analytical performance model for the network will be presented in Section 3. To help in the developing of the model, some useful properties of the network in the presence of hot spot traffic are presented in the next section.

2.3 Network Properties

In the case of uniform traffic, the data rates at all the links between any two stages are the same [15]. In the presence of hot spot traffic, data rates at the links connecting two stages are not the same. In this section, some definitions and important properties of the network under hot-spot conditions are presented. The network properties are simple to prove and can be verified from Figure 1 which shows an Omega network for $N = 8$ with MM_4 being the hot memory module. The properties are required for developing analytical models for hot spot traffic pattern.

Definitions:

- Requests directed to the hot memory module are called *hot requests*.
- The links which carry hot requests are called *hot links*. All SEs connected to hot links are called *hot SEs*. The hot links and SEs in Figure 1 are shown as thick lines and boxes respectively.

Properties:

- All the hot output links of a stage are either the lower outputs or the upper outputs of SEs at that stage.
- Both the inputs to an SE at S_i, $2 \leq i \leq n$ are either the lower or the upper outputs of SEs at S_{i-1}.
- The hot SEs and hot links form a bipartite "hot tree" rooted at the hot memory. The processors form the leaves, and the hot SEs and hot links form the vertices and edges respectively of the tree. There are 2^n paths of this tree – the paths correspond to the paths traversed by hot requests generated by the processors.
- The data rates at the output links of a hot SE are different. The hot output link will have a higher data rate than the cold output link.
- The data rates at the output links of a cold SE are the same.
- Both output links of an SE cannot be hot – either both of them are cold (in the case of a cold SE), or one hot and one cold (in the case of a hot SE).
- Irrespective of cold or hot, both inputs to an SE have the same data rate.

- The number of hot SEs at a stage reduce by half at succeeding stages, and the number of hot links from a stage also reduce by half at succeeding stages. The number of hot SEs at S_i is $2^n/2^i$.

- The inputs to an SE are either hot links or cold links.

- There are i different data rates at the input links of S_i.

3 Performance Analysis

The performance measure considered in this paper is the average bandwidth of the network. The average bandwidth is defined as the average number of packets arriving at the output of the network during a stage cycle. The analytical model will be presented first, and will be followed by a description of the simulation technique.

3.1 Analytical model

A Markov chain model will be developed by assuming that the SEs in a stage are statistically independent in case of uniform traffic, and assigning states to the buffers of an SE. Previous models described in the literature lack the ability to store information regarding the output link of an SE for which a packet was blocked. In this section, we present our 4-state model which memorizes the history of blocking. The state of a buffer in an SE at the beginning of a cycle is represented by one of the possible states of the buffer.

State 0: The buffer is empty.

State n: The buffer contains a new packet which arrived during the previous cycle. This packet will try to forward itself during the current cycle.

State b_u: The buffer contains a packet which is destined for the upper output of the SE and it has been blocked for one or more times in the previous cycles.

State b_l: The buffer contains a packet which is destined for the lower output of the SE and it has been blocked for one or more times in the previous cycles.

For modeling purposes, we split the stage cycle time τ into two phases as in [7, 24, 21].

- In the *first phase* of the cycle (τ_1), the availability of buffer space at the next stage along the destined path of a packet in the current buffer is determined; the packet is informed whether it may go to the next stage or should stay in the current buffer.

- In the *second phase* (τ_2), packets may move forward one stage if the next stage buffers are ready to accept them.

The whole process is repeated in every stage cycle. However, in practice the two phases in a buffer overlap, and enqueuing and dequeuing of packets in a buffer occur simultaneously. We call the state of the buffer at the beginning of phase 1, at the end of phase 1, and at the end of phase 2 of a stage cycle as *initial* state, *intermediate* state, and *final* state respectively. The state probabilities, the routing and blocking probabilities, and the acceptance probabilities are defined next followed by derivation of expressions to determine the bandwidth of the network.

The initial (p_x) and intermediate ($\tilde{p}_x$) state probabilities of a buffer at stage k during the t-th stage cycle are defined as follows.

$p_x(t, k)$ = Pr[a buffer at stage k is in state x at the beginning of the t-th stage cycle, $x \in \{0, n, b_u, b_l\}$].

$\tilde{p}_x(t, k)$ = Pr[a buffer at stage k is in state x at the end of phase 1 of the t-th stage cycle, $x \in \{0, b_u, b_l\}$].

The routing probability of a packet is defined as the probability that a packet in a buffer of an SE can be routed to the next stage. It depends on the states of the other buffers in that SE and the buffer in the SE at the next stage. The routing and blocking probabilities will be denoted by r and $\bar{r}$ respectively and are defined as follows.

$r_x^u(t, k)$ = Pr[a packet in a buffer of an SE at stage k is *able* to move forward to the *upper* output link of the SE during phase 1 of the t-th stage cycle, given that the buffer was in state x at the beginning of the cycle, $x \in \{0, n, b_u, b_l\}$].

$r^l_x(t,k)$ = Pr[a packet in a buffer of an SE at stage k is *able* to move forward to the *lower* output link of the SE during phase 1 of the t-th stage cycle, given that the buffer was in state x at the beginning of the cycle, $x \in \{0, n, b_u, b_l\}$].

$\bar{r}^u_x(t,k)$ = Pr[a packet in a buffer of an SE at stage k is *unable* to move forward to the *upper* output link of the SE during phase 1 of the t-th stage cycle, given that the buffer was in state x at the beginning of the cycle, $x \in \{0, n, b_u, b_l\}$].

$\bar{r}^l_x(t,k)$ = Pr[a packet in a buffer of an SE at stage k is *unable* to move forward to the *lower* output link of the SE during phase 1 of the t-th stage cycle, given that the buffer was in state x at the beginning of the cycle, $x \in \{0, n, b_u, b_l\}$].

The probability that a packet in a buffer after winning any contention in the SE can be forwarded to the next stage depends on whether the next stage buffer is able to accept the packet. The acceptance probabilities (α) of a buffer are defined as follows.

$\alpha(t,k)$ = Pr[a buffer in an SE at stage k is *able to accept* a packet during phase 2 of the t-th stage cycle].

$\alpha_x(t,k)$ = Pr[a buffer in an SE at stage k is *able to accept* a packet during phase 2 of the t-th stage cycle, given that the buffer was in state x at the beginning of the stage cycle].

The traffic rate at a link between two stages is given by the probability that a packet will be routed through the link. This depends on whether a packet is offered to the next stage through the link and the next stage is able to accept it. The probability of offering a packet to the next stage and the link traffic will be denoted by q and ρ respectively

$q(t,k)$ = Pr[a packet is *offered* to a buffer of an SE at stage k during phase 2 of the t-th stage cycle].

The offered packet is received by the buffer at stage k if it was in state 0 at the end of phase 1 of the t-th stage cycle. This may happen if the buffer at stage k was in state 0 at the start of phase 1, or in any of the states n, b_u, or b_l at the start of phase 1 and the packet was successfully routed to stage $k+1$ during phase 1.

$$\rho^u(t,k) = \Pr[\text{a packet is received on the upper input buffer of an SE at stage } k \text{ during phase 2 of the } t\text{-th stage cycle}].$$

$$\rho^l(t,k) = \Pr[\text{a packet is received on the lower input buffer of an SE at stage } k \text{ during phase 2 of the } t\text{-th stage cycle}].$$

The following two lemmas will be used in deriving the expressions for the different state and transition probabilities.

Lemma 1: When the two buffers of an SE at stage k contain a new and a blocked packet at the start of the t-th stage cycle, then the buffer at stage $k+1$ for which the blocked packet (at stage k) is destined cannot be in state 0.

Proof: If the buffer at stage $k+1$ is in state 0 at the beginning of the t-th stage cycle, then there is no reason why the blocked packet at stage k could not be forwarded during phase 2 of cycle $t-1$. Therefore, the buffer at stage $k+1$ must have either

- received a packet from stage k during phase 2 of cycle $t-1$, in which case it will be in state n, or
- received a packet from stage k during a cycle earlier than $t-1$ and the packet could not be forwarded, in which case it will be in state b_u or b_l.

Lemma 2: At the final stage of the network, both the buffers of an SE cannot be in the same (either b_u or b_l) blocked state during any cycle.

Proof: Since there is no output blocking at the final stage of the network as mentioned in Section 2.2, the above conditions cannot happen.

Figure 2 shows the possible state transitions from the initial states of a buffer (beginning of phase 1) to intermediate states (end of phase 1), and from intermediate states to final states (end of phase 2). The probabilities of the different state transitions are also shown in the figure. The initial and intermediate state probabilities can be related as follows.

Intermediate state probabilities

$$\begin{aligned}\tilde{p}_0(t,k) &= p_0(t,k) + p_n(t,k)[r_n^u(t,k) + r_n^l(t,k)] + \\ &\quad p_{b_u}(t,k)r_{b_u}^u(t,k) + p_{b_l}(t,k)r_{b_l}^l(t,k) \qquad (1)\\ \tilde{p}_{b_u}(t,k) &= p_n(t,k)\bar{r}_n^u(t,k) + p_{b_u}(t,k)\bar{r}_{b_u}^u(t,k) \qquad (2)\\ \tilde{p}_{b_l}(t,k) &= p_n(t,k)\bar{r}_n^l(t,k) + p_{b_l}(t,k)\bar{r}_{b_l}^l(t,k) \qquad (3)\end{aligned}$$

State probabilities

$$\begin{aligned}p_0(t+1,k) &= \tilde{p}_0(t,k)[1-q(t,k)] \qquad (4)\\ p_n(t+1,k) &= \tilde{p}_0(t,k)q(t,k) \qquad (5)\\ p_{b_u}(t+1,k) &= \tilde{p}_{b_u}(t,k) \qquad (6)\\ p_{b_l}(t+1,k) &= \tilde{p}_{b_l}(t,k) \qquad (7)\end{aligned}$$

Acceptance probabilities

The probability that a buffer at stage k can accept a packet during cycle t can be expressed in terms of the routing and intermediate state probabilities as given below.

$$\begin{aligned}\alpha_n(t,k) &= r_n^u(t,k) + r_n^l(t,k) \qquad (8)\\ \alpha_{b_u}(t,k) &= r_{b_u}^u(t,k) \qquad (9)\\ \alpha_{b_l}(t,k) &= r_{b_l}^l(t,k) \qquad (10)\\ \alpha(t,k) &= \tilde{p}_0(t,k) \qquad (11)\end{aligned}$$

Equation 8 results from the fact that a buffer in an SE containing a new packet can only accept another packet if the packet can be routed to either the upper or the lower output link of the SE. Similarly, the acceptance probability of a buffer in state b_u (equation (9)) depends on whether the packet can be routed to the upper output link; there is no chance that it can be routed to the lower output link.

Routing and blocking probabilities

In order to calculate the probability of successfully routing a packet in an SE to the next stage, we need to determine the probability of the packet winning any possible contention in the SE and the probability of the possible states of the next stage buffer. Note that in certain cases the state probability of the next stage buffer depends on the states of the buffers at the present stage as explained below.

When there is *a new packet and a blocked packet* at the two buffers of an SE at stage k during cycle t and both of them are destined for the same output link of the SE, the probability that the buffer at stage $k+1$ is able to accept one of these packets is found as follows. The probability that the buffer at stage $k+1$ is in state n at time t is equal to the probability that it received a packet during cycle $t-1$ from the buffer at stage k which contains the new packet at cycle t. This probability is given by $0.5\rho(t-1, k+1)$. Since the next stage cannot be in an empty state under the above circumstances, the probability that the next stage is in a blocked state (either b_u or b_l) is $1 - 0.5\rho(t-1, k+1)$. For uniform traffic in the network, the probability that it is in state b_u or b_l is equal. Thus the probability that is in state b_u (or b_l) is $[1 - 0.5\rho(t-1, k+1)]/2$. In such a case, let's denote the probability that the buffer at stage $k+1$ can accept a packet by $\beta_1(t, k+1)$ which is given by

$$\begin{aligned}\beta_1(t, k+1) &= 0.5\rho(t-1, k+1)\alpha_n(t, k+1) + \\ &\quad 0.5[1 - 0.5\rho(t-1, k+1)][\alpha_{b_u}(t, k+1) + \alpha_{b_l}(t, k+1)] \qquad (12)\end{aligned}$$

When *both the buffers* in an SE at stage k during cycle t are in *blocked* states, the next stage buffer must be in a blocked state. The probability that the buffer at stage $k+1$ can accept a packet is, therefore, given by

$$\beta_2(t, k+1) = 0.5[1 - 0.5\rho(t-1, k+1)][\alpha_{b_u}(t, k+1) + \alpha_{b_l}(t, k+1)] \qquad (13)$$

With the above probabilities of acceptance, we can express the routing probabilities for the packets at stage k, $1 \le j \le n-1$ during cycle t as follows.

$$r_n^u(t, k) = 0.25p_n(t, k)\alpha(t, k+1) + 0.125p_n(t, k)\alpha(t, k+1) +$$

$$
\begin{aligned}
& 0.5p_0(t,k)\alpha(t,k+1) + 0.25p_{b_u}(t,k)\beta_1(t,k+1) + \\
& 0.5p_{b_l}(t,k)\alpha(t,k+1) \\
= \; & [0.375p_n(t,k) + 0.5p_0(t,k) + 0.5p_{b_l}(t,k)]\alpha(t,k+1) + \\
& 0.25p_{b_u}(t,k)\beta_1(t,k+1) \qquad (14)
\end{aligned}
$$

$$
\begin{aligned}
r_n^l(t,k) = \; & [0.375p_n(t,k) + 0.5p_0(t,k) + 0.5p_{b_u}(t,k)]\alpha(t,k+1) + \\
& 0.25p_{b_l}(t,k)\beta_1(t,k+1) \qquad (15)
\end{aligned}
$$

Given that a buffer is in a new state, the sum of the routing and blocking probabilities of a packet must be equal to one. Therefore,

$$
r_n^u(t,k) + r_n^l(t,k) + \bar{r}_n^u(t,k) + \bar{r}_n^l(t,k) = 1 \qquad (16)
$$

Moreover, for uniform traffic in the network the routing and blocking probabilities of a new packet for the upper or lower output links are the same, i.e.

$$
r_n^u(t,k) = r_n^l(t,k) \qquad (17)
$$

$$
\bar{r}_n^u(t,k) = \bar{r}_n^l(t,k) \qquad (18)
$$

Substituting Equations (17) and (18) in (16), we get

$$
\bar{r}_n^u(t,k) = 0.5 - r_n^u(t,k) \qquad (19)
$$

$$
\bar{r}_n^l(t,k) = 0.5 - r_n^l(t,k) \qquad (20)
$$

Given that a packet at stage k, $1 \leq k \leq n-1$ during cycle t is in a blocked state, the routing and blocking probabilities for the packet can be expressed as follows.

$$
\begin{aligned}
r_{b_u}^u(t,k) = \; & 0.5p_n(t,k)\beta_1(t,k+1) + 0.25p_n(t,k)\beta_1(t,k+1) + \\
& p_0(t,k)\beta_1(t,k+1) + 0.5p_{b_u}(t,k)\beta_2(t,k+1) + \\
& p_{b_l}(t,k)\beta_2(t,k+1) \\
= \; & [0.75p_n(t,k) + p_0(t,k)]\beta_1(t,k+1) + \\
& [0.5p_{b_u}(t,k) + p_{b_l}(t,k)]\beta_2(t,k+1) \qquad (21)
\end{aligned}
$$

$$
r_{b_l}^l(t,k) = [0.75p_n(t,k) + p_0(t,k)]\beta_1(t,k+1) +
$$

$$[0.5p_{b_l}(t,k) + p_{b_u}(t,k)]\beta_2(t,k+1) \tag{22}$$

$$\bar{r}^u_{b_u}(t,k) = 1 - r^u_{b_u}(t,k) \tag{23}$$

$$\bar{r}^l_{b_l}(t,k) = 1 - r^l_{b_l}(t,k) \tag{24}$$

Equations (23) and (24) follow from the fact that a packet in a blocked state for a particular output link can either be routed to that particular output link or has to remain in the same blocked state.

It has been assumed in Section 2.2 that the output links of the network are at least as fast as the internal links of the network, i.e. at the last stage of the network there is no blocking due to the next stage not being able to accept an offered packet. The routing and blocking probabilities are, therefore, different from those at the other stages. The routing and blocking probabilities for the last stage ($k = n$) can be obtained from the equations for stage k, $1 \leq k \leq n-1$, by replacing the next stage acceptance probabilities by unity. The resulting equations for $k = n$ are as follows.

$$\begin{aligned} r^u_n(t,n) &= 0.25p_n(t,n) + 0.125p_n(t,n) + 0.5p_0(t,n) + \\ &\quad 0.25p_{b_u}(t,n) + 0.5p_{b_l}(t,n) \\ &= 0.5 - 0.125p_n(t,n) - 0.25p_{b_u}(t,n) \end{aligned} \tag{25}$$

$$r^l_n(t,n) = 0.5 - 0.125p_n(t,n) - 0.25p_{b_l}(t,n) \tag{26}$$

$$\bar{r}^u_n(t,n) = 0.5 - r^u_n(t,n) \tag{27}$$

$$\bar{r}^l_n(t,n) = 0.5 - r^l_n(t,n) \tag{28}$$

$$r^u_{b_u}(t,n) = 1 - 0.25p_n(t,n) \tag{29}$$

$$r^l_{b_l}(t,n) = 1 - 0.25p_n(t,n) \tag{30}$$

$$\bar{r}^u_{b_u}(t,n) = 1 - r^u_{b_u}(t,n) \tag{31}$$

$$\bar{r}^l_{b_l}(t,n) = 1 - r^l_{b_l}(t,n) \tag{32}$$

Traffic rates at the links

In order to determine the bandwidth of the network, we need to determine the traffic rates at the output links of the network. The rates will be the same for all output links in the case of uniform traffic, but will be different in the presence of hot spot traffic. The traffic rate at any link connecting the SEs between stages $k-1$ and k (or the load applied to any stage k) can be obtained by finding the probability that a packet is received in a buffer of an SE at stage k from the SE at stage $k-1$. A packet can be received in a buffer at stage k during cycle t if at least one of the buffers in the SE at stage $k-1$ had a packet in either state n, b_u, or b_l at the end of phase 2 of cycle $t-1$ and was able to route the packet during cycle t. Therefore, the traffic rate at an input link of an SE at stage k which receives packets from the upper output links of SEs at stage $k-1$ can be given by

$$\rho^u(t,k) = 2[p_n(t,k-1)r_n^u(t,k-1) + p_{b_u}(t,k-1)r_{b_u}^u(t,k-1)] \tag{33}$$

The multiplying factor of 2 in equation (33) comes from the fact that an input buffer of an SE at stage k can receive a packet from any of the two buffers of the SE at stage $k-1$. The input traffic rate for SEs at stage k receiving packets from lower output links of SEs at stage $k-1$ is given by

$$\rho^l(t,k) = 2[p_n(t,k-1)r_n^l(t,k-1) + p_{b_l}(t,k-1)r_{b_l}^l(t,k-1)] \tag{34}$$

For uniform traffic in the network, $\rho^u(t,k) = \rho^l(t,k) = \rho(t,k)$, but are different for hot spot traffic in the network.

The probability that a packet is offered to a buffer of an SE at stage k is the ratio of the probability that a packet is received by the buffer to the probability that the buffer can accept the packet.

$$q(t,k) = \rho(t,k)/\alpha(t,k) \tag{35}$$

The iterative method described in [7] has been used to solve the above set of equations. Starting with an initial value of the load applied to the network ($q(t,1)$) and assuming all buffers being empty, the above set of equations are solved iteratively until time independent steady state values are obtained for the state, acceptance, routing, and blocking probabilities at the different stages. Steady state value of any variable $x(t,k)$ will be represented as $x(k)$.

The average bandwidth of the network is obtained by summing the data rates at all the output links of the network. For uniform traffic, the sum is given by $N\rho(n+1)$, where $\rho(n+1)$ is obtained from the steady state value of equation (33) for $k = n+1$.

3.2 Simulation model

To validate our proposed analytical model described in the previous section, stochastic simulation was carried out both for uniform and hot spot traffic. The assumptions mentioned in Section 2.2 were implemented in the simulator as follows.

- Each processor generates a packet independently at each stage cycle with probability $q(1)$.
- The destination of a generated packet is taken from a uniform random number generator in case of uniform traffic, and in case of hot spot traffic, from a non-uniform random number generator which generates requests according to the hot spot probability distribution mentioned in Section 2.2.
- In case of routing conflicts among packets in an SE, a packet is selected at random by another random number generator.
- The number of requests arriving at the output of the network per stage cycle was averaged over a large number of cycles (typically 50,000) to obtain the average bandwidth. Moreover, the outputs for the first 500 cycles were discarded to allow the network to reach a steady state.

4 Results

The impact of simplifying assumptions on the results obtained from the proposed model will be shown in this section by comparison with results from simulation and existing models. As mentioned earlier, we have developed hot spot traffic versions of the original models of Jenq and Theimer. Results obtained from the different models for uniform and hot spot traffic in a 1024×1024 Omega network are given below.

Jenq's original model was modified for hot spot traffic by accounting for the non-uniform distribution of packets at the hot SEs, and the fact that the i-th stage consists of i different types of SEs. Figure 3 shows the percentage errors in bandwidth obtained from Jenq's model and simulation in the presence of hot spot traffic. It is observed that the errors increase drastically with an increase in the network traffic rate or/and probability of hot spot requests. Since the model does not account for blocked packets, the errors increase with an increase in the blocking at the SEs. Note that an increased level of hot spot traffic and/or network traffic rate cause increased blocking at the switches. Increased network traffic causes blocking at all the SEs uniformly, while increased hot spot traffic causes blocking at the hot SEs giving rise to tree saturation.

We have also modified Theimer's original model by taking account of the non-uniform packet distribution at the hot SEs, and the different data rates at the SEs at a stage in the case of hot spot traffic. Bandwidths obtained from the modified model and the simulator are shown in Figure 4 for $N = 1024$ and $h = 0.005972$. Since blocked packets are taken into account partially, the results are closer to simulation than those from Jenq's model. Since Theimer's model does not completely account for a blocked packet, the errors have been found to rise with increased blocking in the network resulting from hot spot traffic. Though the model takes into account the blocked packets, it does not store the history of blocking, i.e. the output link for which a packet was blocked. It is important to store this history because a blocked packet always hunts for the same output link during successive cycles.

Uniform traffic causes less blocking in the SEs than hot spot traffic. Consequently, the accuracy of Jenq's and Theimer's models are better for uniform traffic than for hot spot traffic as is evident from Figure 5 which is a comparison of bandwidths obtained from simulation, Jenq's model, Theimer's model and our proposed model under uniform traffic conditions. It is observed that Theimer's model and our proposed model produce similar results for uniform traffic. This is because the *inaccuracy in Theimer's model does not show up until there is significant blocking in the network* as illustrated below.

Figure 6 shows the bandwidths obtained from simulation, Jenq's model, Theimer's model, and our proposed model *in the presence of hot spot traffic* for $N = 1024$ and $h = 0.005972$. Because of severe blocking in the network (tree saturation) in the presence of hot spot traffic, our proposed model produces results which are close to simulation, and are *significantly better than those obtained from Jenq's or Theimer's models.* The reason for improved results from our proposed model is the fact that the model memorizes the history of a blocked packet, whereas Theimer's model does not memorize the history and Jenq's model does not even consider blocked packets.

The effect of taking a rigorous account of blocking in our model is reflected in the increased accuracy in results in the presence of significant blocking in the network. In the presence of hot spot traffic, Theimer's model performs as good as the proposed model only when the blocking in the network is insignificant. As an example, Theimer's model produces good results for $0 \leq q(t,1) \leq 0.2$ and $h = 0.005972$ (see Figure 6) because of insignificant blocking in this range of network traffic rate. However, if the hot spot probability increases, Theimer's model produces large errors even for the above range of traffic rate.

All the models have been solved iteratively using a Sun Sparcstation. For a set of points, our model requires about 50 secs of execution time on such a workstation.

5 Conclusion

An analytical model permits a fast and inexpensive method of performance evaluation of interconnection networks. Previous analytical models have, in most cases, assumed uniform traffic in the network. Recent studies have shown that traffic in a network is non-uniform in case of a hot spot. A new analytical model for a single buffered Omega network, based on the Markov chain, has been presented in this paper. The model allows the investigation of performance of MINs in the presence of hot spot traffic pattern. Modifications of previous models to hot spot traffic did not produce accurate results. In the presence of hot spot traffic, the proposed model produces significantly better results than previous models. For uniform traffic (which is a special case of hot spot traffic), the results obtained from the proposed model are as good as results obtained from the best previous analytical model. The reason behind the proposed model's higher accuracy in the presence of hot spot traffic is its ability to take a rigorous account of blocking at the SEs, by memorizing the output link of an SE for which a packet was blocked. Uniform traffic does not cause as much blocking as hot spot traffic, due to which the results obtained from the proposed model are similar to those obtained from previous models in the case of uniform traffic. The inaccuracy of previous models becomes apparent in the presence of considerable blocking in the network.

The novelty of the proposed model has been shown by deriving equations for the uniform traffic case. The two buffers at an SE have been considered to be statistically indistinguishable. Equations for the proposed model can can be derived for the hot spot traffic case by separately considering the two buffers of an SE. Therefore, two sets of equations (corresponding to the two buffers of an SE) are required, and the equations are properly linked together taking into account the properties of the network with a single hot spot as described in Section 2.3. The proposed model can be extended to other types of non-uniform traffic patterns [22], and to Banyan type networks which are characterized by a unique path from source to destination. The delay experienced by a packet depends on the number of hot switches it encounters from the source to the destination.

References

[1] M. Atiquzzaman and M.S. Akhtar. Effect of hot spots on the performance of multistage interconnection networks. In *FRONTIERS 92: The Fourth Symposium on the Frontiers of Massively Parallel Computation*, pages 504–505, Virginia, October 19-21, 1992.

[2] M. Atiquzzaman and M.S. Akhtar. Effect of non-uniform traffic on the performance of unbuffered multistage interconnection networks. *IEE Proceedings - Computer and Digital Techniques*, 141(3):169–176, May 1994.

[3] L.N. Bhuyan. An analysis of processor-memory interconnection networks. *IEEE Transactions on Computers*, C-34(3):279–283, March 1985.

[4] D.M. Dias and J.R. Jump. Analysis and simulation of buffered Delta networks. *IEEE Transactions on Computers*, C-30(4):271–282, April 1981.

[5] A. Gottlieb, R. Grishman, C.P. Kruskal, K.P. McAuliffe, L. Rudolph, and M. Snir. The NYU ultracomputer-Designing and MIMD shared memory parallel computer. *IEEE Transactions on Computers*, C-32(2):175–189, February 1983.

[6] S.H. Hsiao and R.Y. Chen. Performance analysis of single-buffered multistage interconnection networks. In *Third IEEE Symposium on Parallel and Distributed Processing*, pages 864–867, December 1-5, 1991.

[7] Y-C. Jenq. Performance analysis of a packet switch based on single-buffered Banyan network. *IEEE Journal on Selected Areas in Communications*, SAC-1(6):1014–1021, December 1983.

[8] H.S. Kim and A. Leon-Garcia. Performance of buffered Banyan networks under nonuniform traffic patterns. *IEEE Transactions on Communications*, 38(5):648–658, May 1990.

[9] H.S. Kim, I. Widjaja, and A. Leon-Garcia. Performance of output-buffered Banyan networks with arbitrary buffer sizes. In *IEEE INFOCOM '91: Conference on Computer Communications*, pages 701–710, Bal Harbour, Florida, April 9-11, 1991.

[10] C.P. Kruskal and M. Snir. The performance of multistage interconnection networks for multiprocessors. *IEEE Transactions on Computers*, C-32(12):1091–1098, December 1983.

[11] T. Lang and L. Kurisaki. Nonuniform traffic spots (NUTS) in multistage interconnection networks. *Journal of Parallel and Distributed Computing*, 10(1):55–67, September 1990.

[12] D.H. Lawrie. Access and alignment of data in a array processor. *IEEE Transactions on Computers*, C-24(12):1145–1155, December 1975.

[13] G. Lee, C.P. Kruskal, and D.J. Kuck. The effectiveness of combining in shared-memory parallel computers in the presence of hot spots. In *1986 International Conference on Parallel Processing*, pages 35–41, August 1986.

[14] T. Lin and L. Kleinrock. Performance analysis of finite-buffered multistage interconnection networks with a general traffic pattern. In *1991 ACM SIGMETRICS Conference on Measurement and Modeling of Computer Systems*, pages 68–78, San Diego, CA, May 21-24, 1991.

[15] J.H. Patel. Performance of processor-memory interconnections for multiprocessors. *IEEE Transactions on Computers*, C-30(10):771–780, October 1981.

[16] N.M. Patel and P.G. Harrison. On hot-spot contention in interconnection networks. In *1988 ACM SIGMETRICS Conference on Measurement and Modeling of Computer Systems*, pages 114–123, May 24-27, 1988.

[17] G.F. Pfister, W.C. Brantley, D.A. George, S.L. Harvey, W.J. Kleinfelder, K.P. McAuliffe, E.A. Melton, V.A. Norton, and J. Weiss. The IBM Research Parallel Processor prototype (RP3): Introduction and architecture. In *1985 International Conference on Parallel Processing*, pages 764–771, St. Charles, Illinois, August 1985.

[18] G.F. Pfister and V.A. Norton. Hot spot contention and combining in multistage interconnection networks. *IEEE Transactions on Computers*, C-34(10):943–948, October 1985.

[19] S.L. Scott and G.S. Sohi. The use of feedback in multiprocessors and its applications to tree saturation control. *IEEE Transactions on Parallel and Distributed Systems*, 1(4):385–398, October 1990.

[20] S. Thanawastien and V.P. Nelson. Interference analysis of shuffle/exchange networks. *IEEE Transactions on Computers*, C-30(8):545–556, August 1981.

[21] T.H. Theimer, E.P. Rathgeb, and M.N. Huber. Performance analysis of buffered Banyan networks. *IEEE Transactions on Communications*, 39(2):269–277, February 1991.

[22] L.T. Wu. Mixing traffic in a buffered Banyan network. In *9th Data Communications Symposium*, pages 134–139, September 1985.

[23] P.C. Yew, N.F. Tzeng, and D.H. Lawrie. Distributing hot-spot addressing in large-scale multiprocessors. *IEEE Transactions on Computers*, C-36(4):388–395, April 1987.

[24] H. Yoon, K.Y. Lee, and M.T. Liu. Performance analysis of multibuffered packet-switching networks in multiprocessor systems. *IEEE Transactions on Computers*, 39(3):319–327, March 1990.

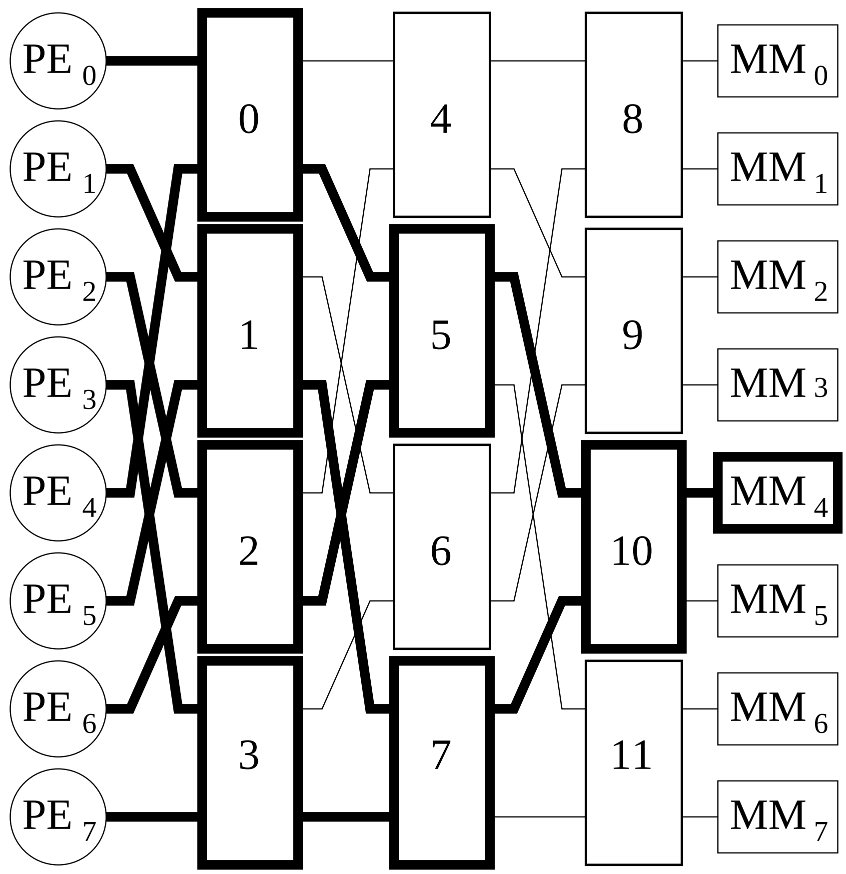

Figure 1: An 8×8 Omega network.

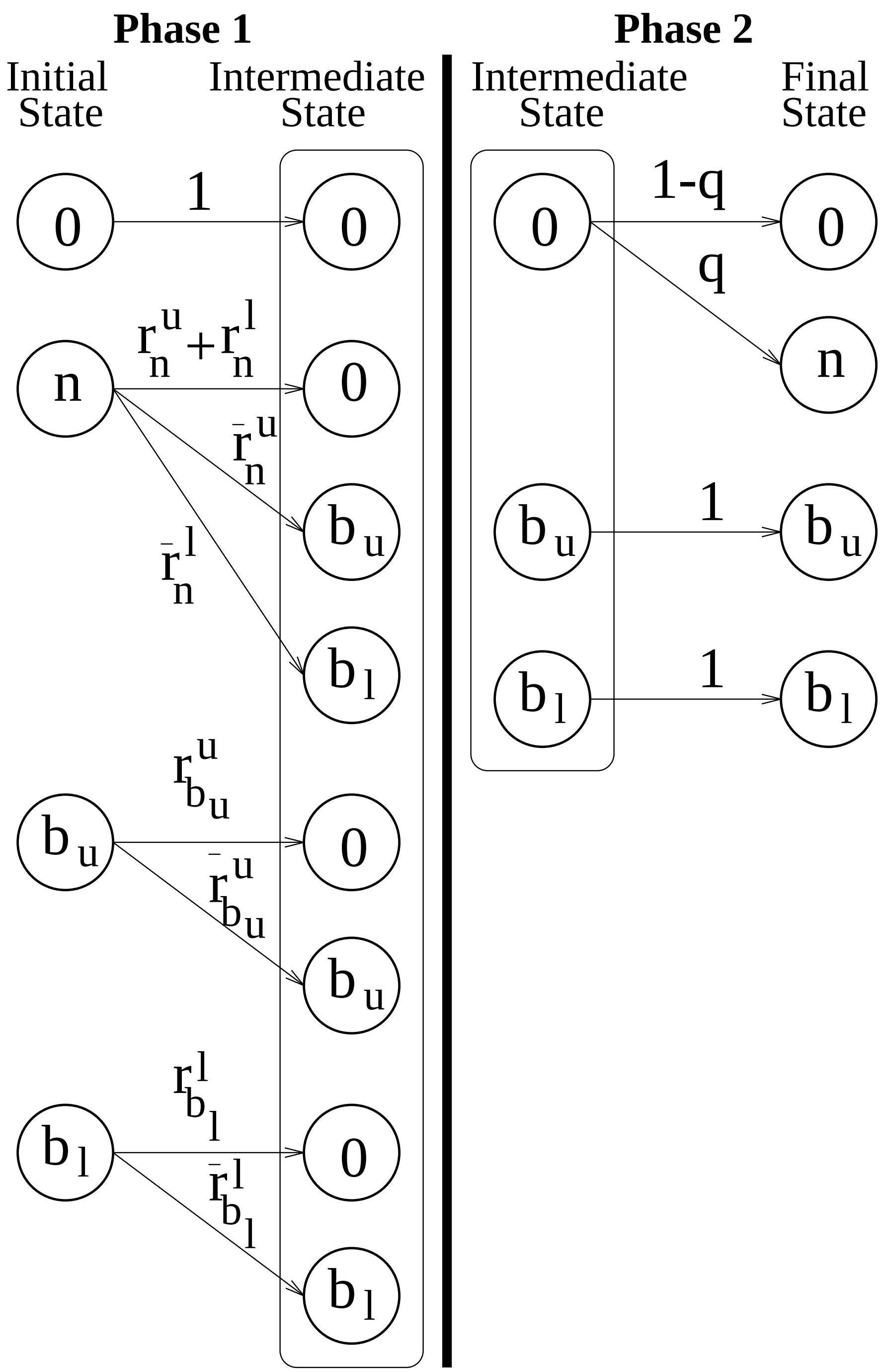

Figure 2: State transitions diagram of a buffer at a switching element.

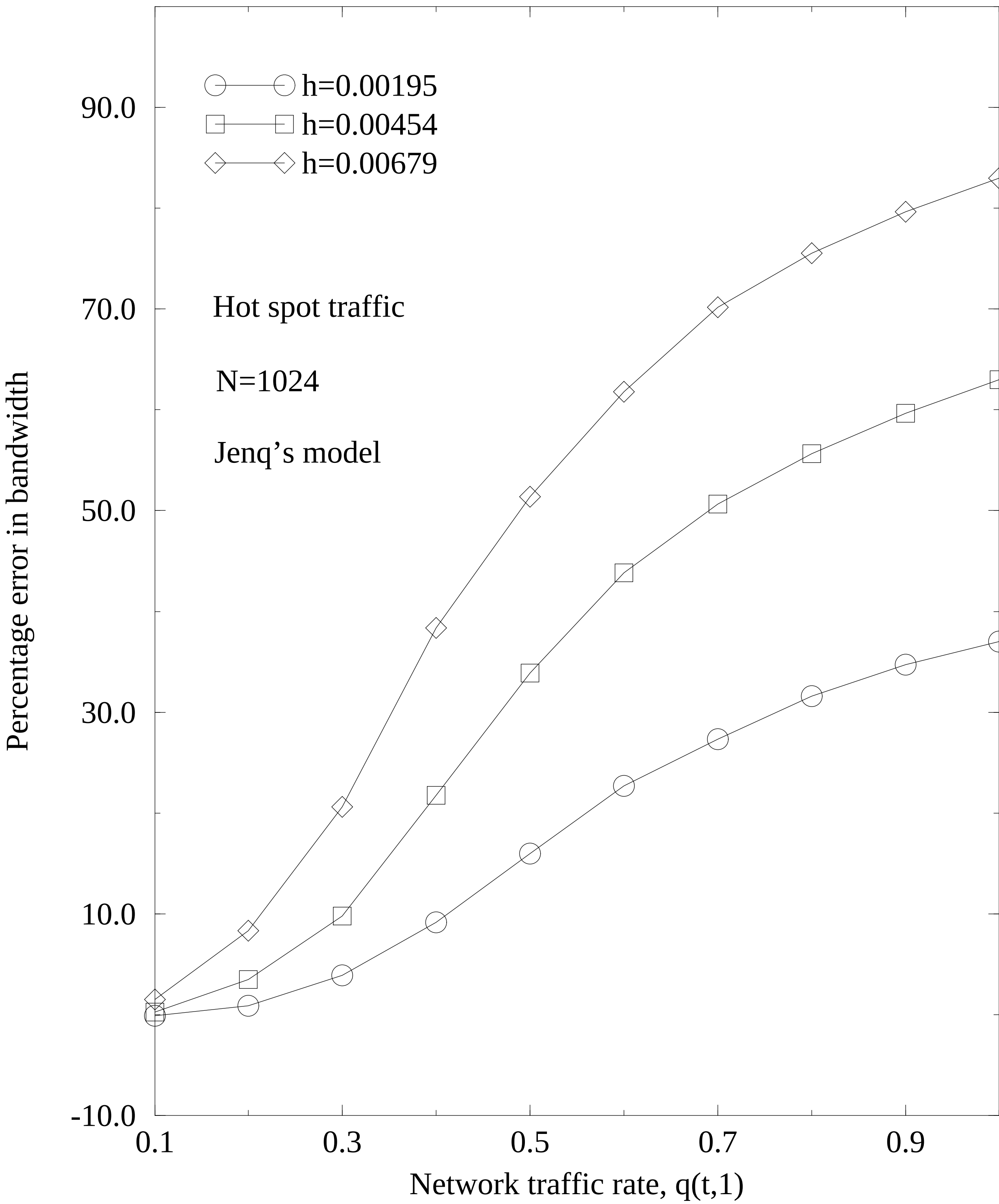

Figure 3: Percentage errors in bandwidth obtained from Jenq's model and simulation for different hot spot probabilities.

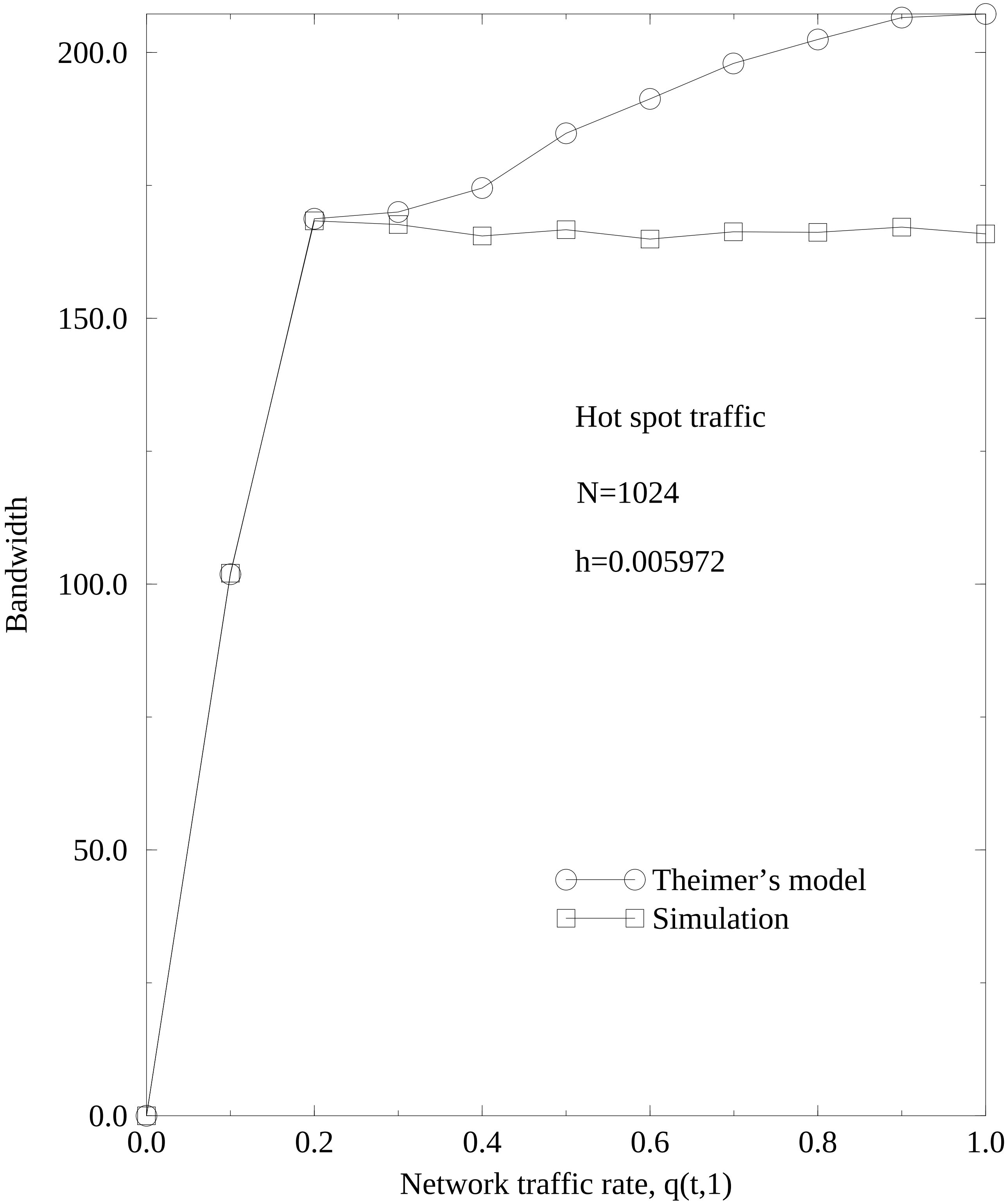

Figure 4: Comparison of bandwidths obtained from Theimer's model and simulation for $h = 0.005972$.

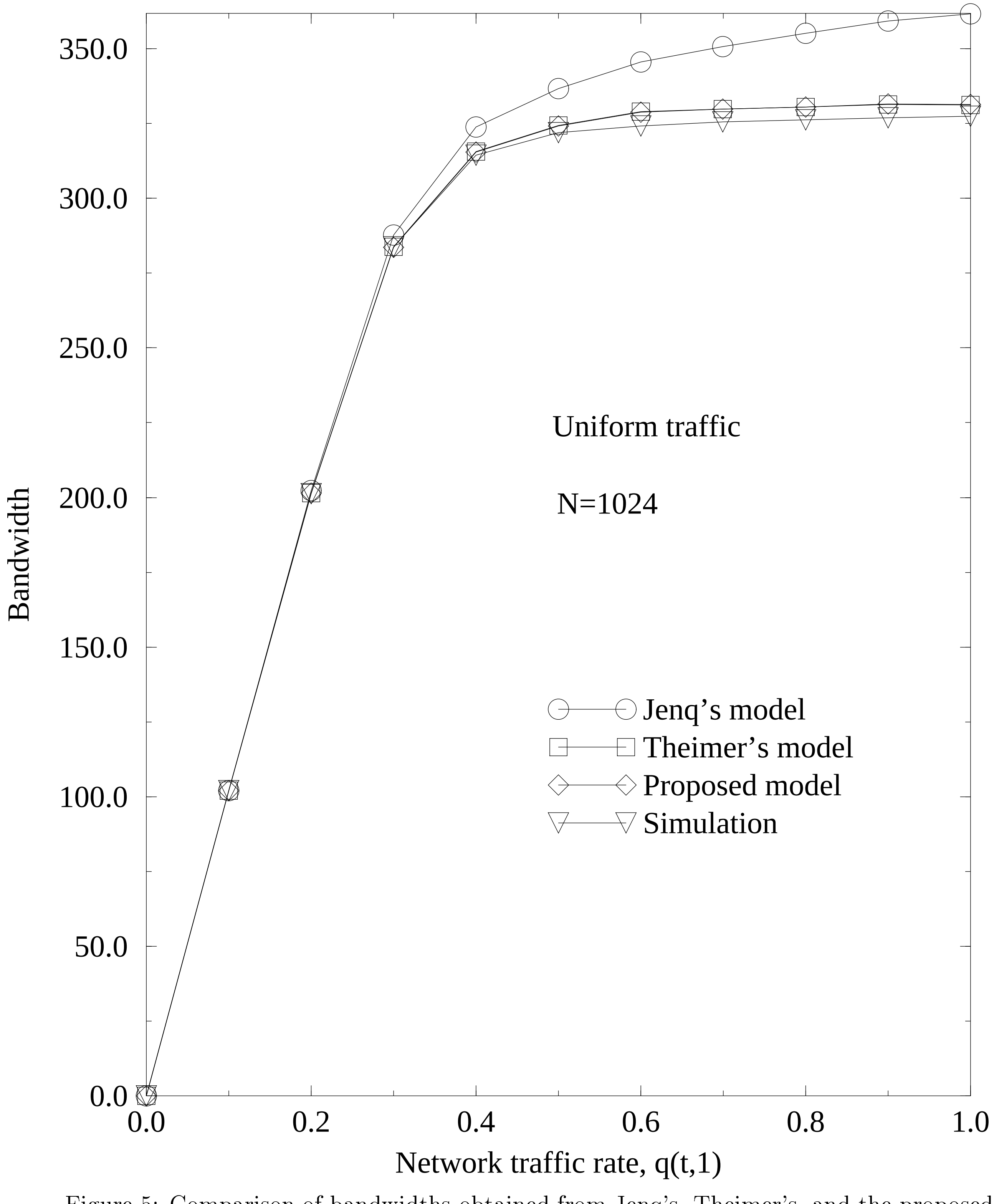

Figure 5: Comparison of bandwidths obtained from Jenq's, Theimer's, and the proposed model in the presence of uniform traffic.

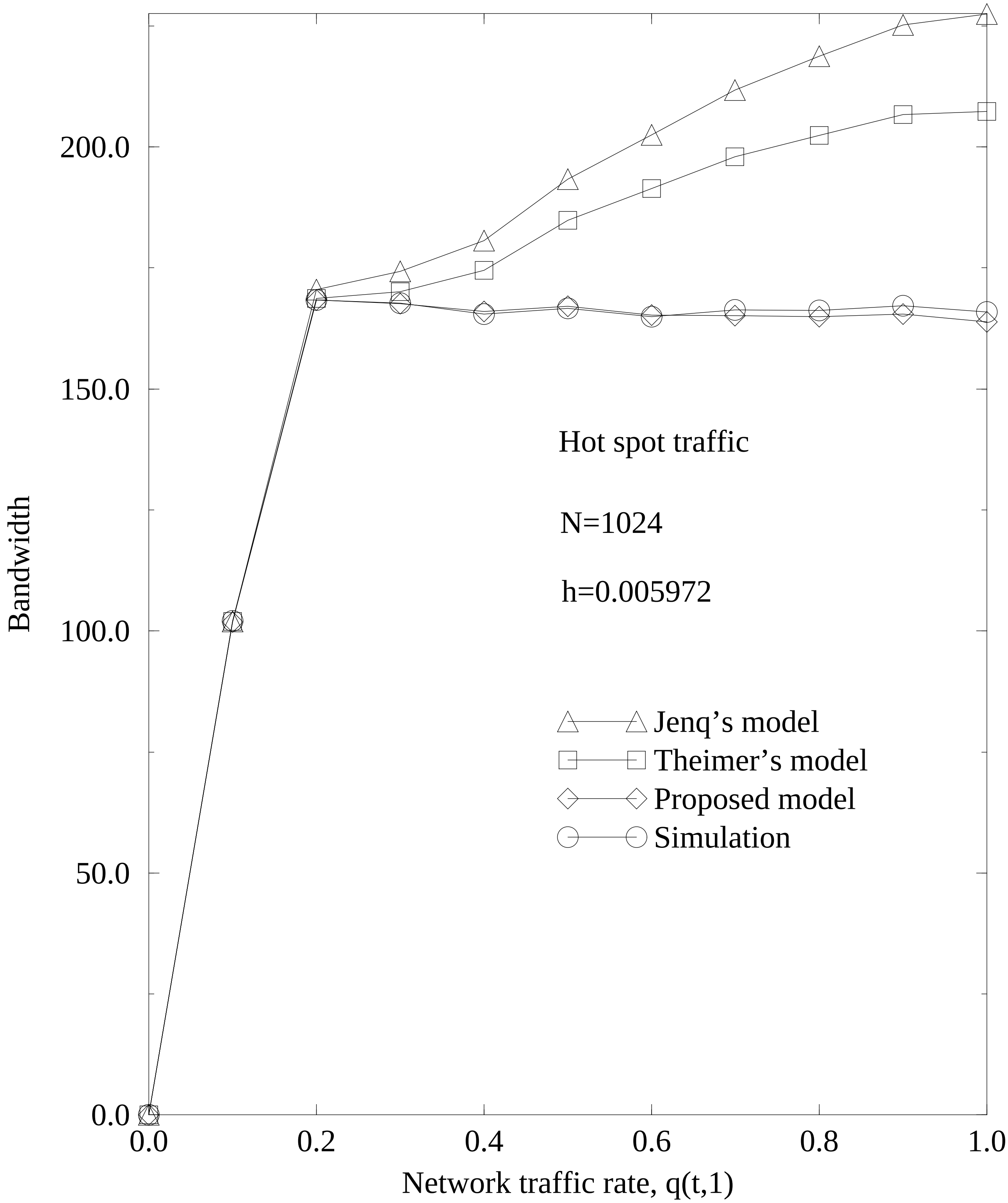

Figure 6: Comparison of bandwidths obtained from Jenq's, Theimer's, and the proposed model for $h = 0.005972$.