6th International Conference on Applied Informatics
Eger, Hungary, January 27–31, 2004.

Investigating Multicasting Traffic in Multistage Interconnection Networks with MOSEL*

Béla Almási[a], Dietmar Tutsch[b]

[a]Institute of Informatics, University of Debrecen
e-mail: almasi@inf.unideb.hu

[b]Computer Science Departement, Technical University of Berlin
e-mail: dietmart@cs.tu-berlin.de

Abstract

Multistage interconnection networks (Banyan networks) are frequently used for performance evaluation of communication systems (e.g. Ethernet and ATM switches). There exist several studies (see [9] and [7]) describing the performance of networks in which packets (or frames) are unicasted (i.e. one switching element (SE) sends the packet to one and only one other SE). A model for MINs in case of packet multicasting (i.e. the SE may send the packet to one or more SE) was introduced in [11].

In this paper a timed Petri net model is used to analyze the packet traffic in multistage interconnection networks in the case of unicasting and multicasting too. We consider a Banyan network with 2x2-switches and the two cases of complete and partial broadcasting within the switching elements.

We study how effectively the universal modelling environment of MOSEL (see [4]) can be applied for the mentioned systems to produce analytical performance measures. Finally some graphically represented numerical results illustrate the problem in question.

Categories and Subject Descriptors: C.2.3 [Computer-Communication Networks]: Network Operations; I.6.5 [Simulation and Modelling]: Model Development;

Key Words and Phrases: Multistage interconnection network, multicasting.

*The authors are very grateful to Professor Gunter Bolch for his helpful comments. Research is partially supported by Hungarian Scientific Research Fund OTKA T0-34280/2000 and FKFP grant 0191/2001.

1. Introduction

Stochastic modelling of Multistage Interconnection Networks (MINs) provides the performance evaluation of networks, which are frequently used to connect multiprocessor and communication systems (e.g., Ethernet switches). Such systems require high performance of the network. To increase the performance of a MIN, Dias et al. [5] inserted a buffer at each input of the switching elements (SE) and developed a stochastic analytical model to predict its performance. Buffers at each SE allow to store the packets of a message until they can be forwarded to the next stage in the network. In their model, Dias and Jump reduced each stage in the network to one SE of this stage so that it could be mapped onto a Markov chain.

Jenq [6] introduced a model with lower complexity than that of Dias and Jump by considering only one input port of an SE per stage to model the complete stage. Yoon et al. [15] extended Jenq's model by using arbitrary buffer lengths in the network and arbitrary SE sizes. Atiquzzaman et al. [2] and Zhou et al. [17] examined nonuniform traffic like hot spot traffic. Cut-through switching was taken into account by Widjaja et al[14]. On the other hand, there are a few investigations on multicast routing in MINs [8] and on the structure of multicast ATM switches.

Tutsch et al. [10] extended Jenq's model such that the stochastic model additionally copes with performance analysis of a network with multicasting. Multicasting includes the two special cases of unicasting and broadcasting of messages. The model uses store and forward routing and the backpressure mechanism.

A system of equations was set up manually for performance estimation. During the set up some rules emerged to build such a system. These rules were extended for automatic generation of systems of equations [13], which cope with the multicast performance analysis of MINs consisting of switching elements larger than 2×2.

To validate the results of the analytical model, a Petri net description of MINs based on 2×2 switching elements was established [10]. Due to state space explosion, just simulation was available to receive results.

In this paper, a further method of a stochastic MIN modelling is investigated. MOSEL language developed at University of Erlangen is used to establish a model similar to the previously mentioned Petri net description.

The paper is organized as follows. The stochastic model and the structure of the investigated MINs are presented in Section 2. In Section 3 we consider a short overview of the MOSEL system, which was used to calculate analytical results for the described MIN models. Some numerical results are presented in Section 4, and finally Section 5 gives a conclusion.

2. MIN Model and Behaviour

The following generated analytical model allows to determine the performance of internally clocked $N \times N$ MINs (i.e., a MIN with N input and N output ports) consisting of 2×2 switches with $n = \log_2 N$ stages. The structure of an 8×8 MIN with $n = 3$ stages can be seen on Figure 1.

Internal clocking results in synchronously operating switches. In each stage k ($0 \leq k \leq n-1$), there is a FIFO buffer of size $m_{max}(k)$ in front of each switch input. The packets are routed by store and forward routing from a stage to its succeeding one by backpressure mechanism. Multicasting is performed by copying the packets within the switches (SEs).

The assumptions for the model are the same as those of Jenq's Model I [6]. Most analytical MIN models deal with those assumptions to reduce the problem complexity in such way that an analytical model can be established. Uniform traffic and the independence of the packets are assumed according to the following assumptions:

- The traffic load of all inputs of the network is equal.
- All packets have the same size (like in ATM).
- Their destination outputs are distributed uniformly. That means every output of the network is with equal probability one of the destinations of a packet.
- Conflicts between packets are solved randomly with equal probabilities.
- Packets are removed from their destinations immediately after arrival.
- Routing is performed in pipeline manner. That means the routing process occurs in every stage in parallel.

Due to the conflict resolution policy and the way how the destinations of packets are determined, the model results in a stochastic one.

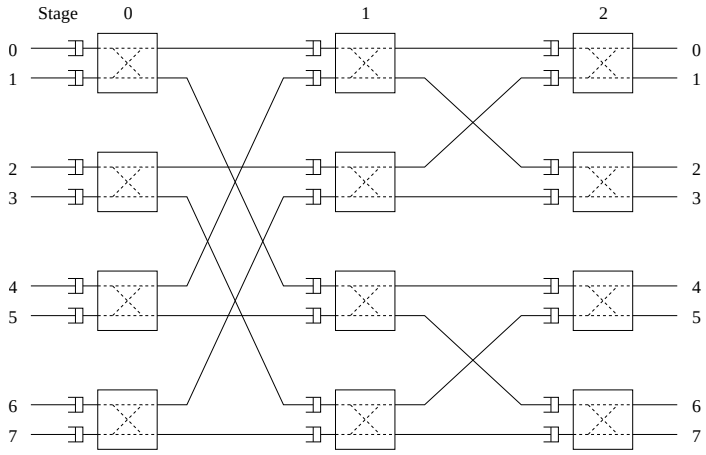

Figure 1: 3-stage 8×8 MIN consisting of 2×2 SEs.

The 2×2 switches can operate with partial or with complete multicasting: Partial multicasting means, that if anyone of the destination output ports or destination buffers is not available, the packet will stay in the present stage and copies will only be transmitted to the available destinations. The transmission to the other destinations is performed later. Complete multicasting means, that the packet will be transmitted to the next stage if and only if both output ports (destination buffers) are available.

As multicasting is considered in this paper, many different assumptions about the shape of the network traffic are possible. The most simple case is to assume that all possible combinations of destination addresses (i.e., each subset of the output ports) are equally distributed for each packet entering the network. This traffic pattern is called traffic_{eqpr}.

A more realistic assumption is that we have a high amount of unicasting and broadcasting and that multicasting to a few and to many destination addresses is more likely than to a medium number of destination addresses. This traffic pattern is called traffic_{smhi}.

The precise definition of those traffic patterns is given in [10]. The patterns influence the multicast behaviour within the switching elements. A newly received packet at a stage can be directed to one (upper or lower), or to the both switching element outputs.

In our model, multicast probabilities are introduced representing the average percentage of packets that are destined to a certain amount of switch outputs: The multicast probability $\omega_{mult}(k)$ denotes the probability that the newly received packet in stage k is directed to $mult$ switch outputs ($mult = 1, 2$). It is obvious, that this probability depends on the number of network outputs that are demanded by the packets arriving at the MIN inputs.

Let $a(i)$ denote the probability that a packet arriving at a MIN input is directed to a destination set of i MIN outputs ($1 \leq i \leq N$). That means $a(i)$ represents the given global network traffic, i.e., traffic_{eqpr} or traffic_{smhi}. In [12], the multicast probability $\omega_{mult}(k)$ of any stage k is calculated for a given multicast traffic pattern $a(i)$. $\omega_{mult}(k)$ influences the resource utilisation (e.g., buffers) and allows to determine the steady state performance measures of the MIN.

Because of the hierarchical structure of the MIN (see Figure 1), the most important task of the modelling is to establish a stochastic model for a 2×2 switching element. Assuming, that the switching times are independent exponentially distributed random variables a Markov chain can be established for the mathematical model of a SE. Unfortunately, the state space of the Markov chain is too large (even in the case of a 2×2 SE), so we will not describe the balance equations. Instead of it we consider the stochastic Petri net model of a 2×2 SE in the case of complete multicasting (see Figure 2). Further details on the SPN model can be read in [10].

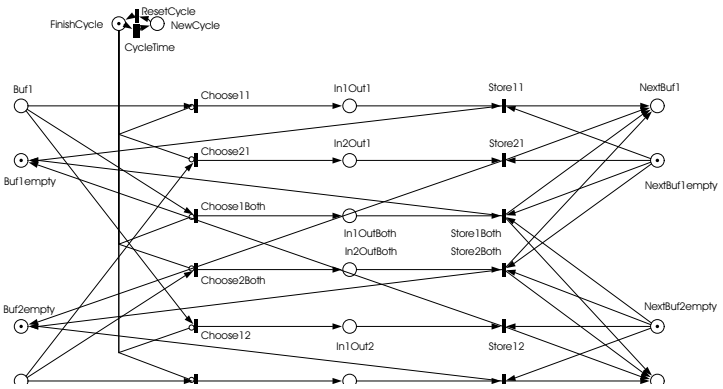

Figure 2: Stochastic Petri net model of 2×2 SEs.

3. The MOSEL Modelling Environment

This section gives a short description of MOSEL - the most important basics of our software tool. MOSEL (MOdelling Specification and Evaluation Language) is a language, developed at the University of Erlangen. The MOSEL system uses a macro-like language (see [4]) tuned especially to describe stochastic Petri nets. The MOSEL programs consist of four parts: the declarations, the node definitions, the transition rules and the results.

In the declaration part we can declare constants and variables. Macro definition is also possible, which allows shortening the program. Here are the most important constants of our implementation:

```
/* ================= Declaration part ===================*/
#define  N  2      // No. of I/O ports; change it to 4, 8, 16 ...
#define  LN 1      // Log2(N), No. of stages; change it with N.
#define  BUFSIZE 1 // The size of the input buffer of the SE.
```

In the node part we define the nodes (i.e. the places in the SPN model) for the system. We can use the constants defined in the declaration part. We can also give initial values for the nodes (i.e. the initial number of tokens in the places). In the following example we define LN pieces of nodes, which shows the active/passive state of a SE in a stage (the SEs work in parallel in a stage):

```
<1..LN>          NODE  sw<#1>[1]          = 0;
```

The transition rule part is the most important part of the MOSEL program, which describes the system's behaviour using FROM ... TO style rules. Please refer to [4] for further details. The result part calculates the output results. The

results are specified by equations in which we can use the word PROB to refer to the steady-state probability of a given state.

We do not intend to analyse the MOSEL source of our implementation in this paper, further details can be received from the authors.

4. Numerical Results

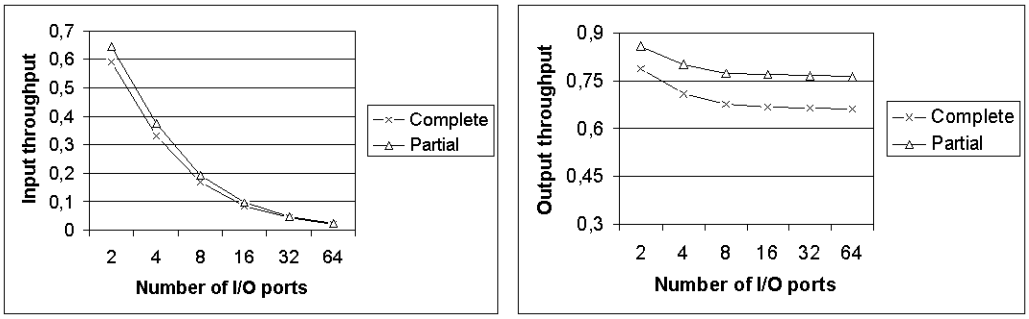

Figure 3: Throughput using complete and partial broadcasting.

We have implemented the earlier described MIN models in MOSEL to get analytical performance measures. The MOSEL program can be used for arbitary sized MINs, (only the above mentioned 3 constants must be modified), but hardware constraints just allowed 2×2 MINs. We also used a simulation program to get results for larger MINs, and also to validate the analytical results for 2x2 switches.

In figure 3 we can compare the input and output throughput depending on the size of the MIN in the case of partial and complete multicasting. As it can be expected the difference between the partial and complete broadcasting at the input side is really small, and the difference is decreasing with the size of the MIN. That is due to the larger amount of multicasting and the backpressure mechanism, which arises a lower input rate. At the output side, the difference is much higher caused by higher throughput rates resulting from multicasting.

Figure 4 shows the relationship between the input buffer size and the input throughput. For small MINs (e.g. the 2x2 MIN) the size of the buffer is not relevant, the throughput measures are the same. As the size of the MIN grows, so increases the influence of the buffer size: greater buffer size produces greater throughput.

In figure 5 an interesting result can be observed: The influence of the multicast distribution (considering traffic_{eqpr} and traffic_{smhi} as mentioned earlier) is quite small on the delay time even in the case of large MINs.

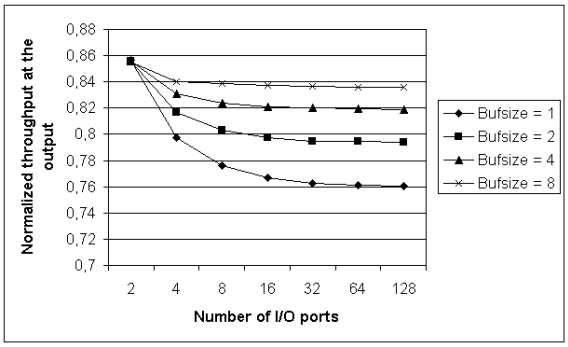

Figure 4: The effect of the buffer length on the throughput.

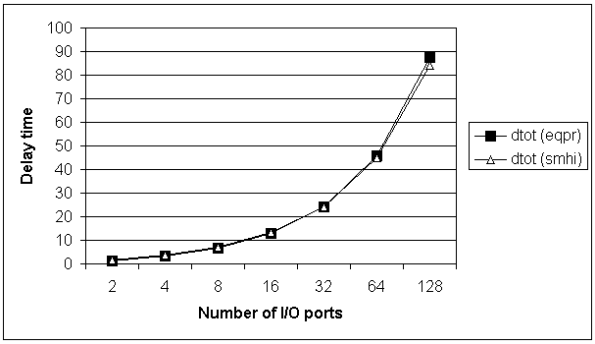

Figure 5: The effect of the multicast distribution on the delay time.

5. Conclusion

In this paper a stochastic Petri net model has been treated to analyse packet traffic in multistage interconnection networks. Also a software tool was introduced (based on MOSEL and SPNP) which can be used to calculate analytical results for the model. The currently available hardware environment allowed to investigate 2x2 MINs (i.e. we can get results for the building-block of a MIN), but we hope, that this MOSEL implementation can be used for larger MINs in the near future, as the computer hardware resources are growing day by day. Finally some numerical examples (generated by the MOSEL program and a simulation tool) illustrated the problem in question.

References

[1] G. A. Abandah and E. S. Davidson, Modeling the communication performance of the IBM SP2. In *Proceedings of the 10th International Parallel Processing Symposium*

(IPPS'96); Hawaii. IEEE Computer Society Press, 1996.

[2] M. Atiquzzaman and M. S. Akhtar, Performance of buffered multistage interconnection networks in a nonuniform traffic environment. *Journal of Parallel and Distributed Computing*, 30(1):52–63, October 1995.

[3] R. Y. Awdeh and H. T. Mouftah, Survey of ATM switch architectures. *Computer Networks and ISDN Systems*, 27:1567–1613, 1995.

[4] K. Begain, G. Bolch, H. Herold, Practical Performance Modeling, Kluwer Academic Publisher, Boston, 2001.

[5] D. M. Dias and J. R. Jump, Analysis and simulation of buffered delta networks. *IEEE Transactions on Computers*, C–30(4):273–282, April 1981.

[6] 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.

[7] Y. Mun and H. Y. Youn, Performance analysis of finite buffered multistage interconnection networks *IEEE Transactions on Computers,* Vol. 43, No. 2, (1994).

[8] R. Sivaram, D. K. Panda, and C. B. Stunkel. Efficient broadcast and multicast on multistage interconnection networks using multiport encoding. *IEEE Transaction on Parallel and Distributed Systems*, 9(10):1004–1028, October 1998.

[9] T. H. Theimer, E. P. Rathgeb and M. N. Huber, Performance analysis of buffered banyan networks, *IEEE Transactions on Communications,* Vol. 39, No. 2, (1994).

[10] D. Tutsch and G. Hommel, Performance of buffered multistage interconnection networks in case of packet multicasting. In *Proceedings of the 1997 Conference on Advances in Parallel and Distributed Computing (APDC'97); Shanghai*, pages 50–57. IEEE Computer Society Press, March 1997.

[11] D. Tutsch and W. Wilhelmi, Leistunganalyse bei Multicasting in gepufferten mehrstufigen Verbindungsnetzweken (in German) *Technical Report 1995-02,* Technical University Berlin, Berlin, (1999).

[12] D. Tutsch and M. Brenner. Multicast probabilities of multistage interconnection networks. In *Proceedings of the 12th European Simulation Symposium 2000 (ESS'00); Hamburg*, pages 554–558. SCS, September 2000.

[13] D. Tutsch and G. Hommel. Generating systems of equations for performance evaluation of buffered multistage interconnection networks. *Journal of Parallel and Distributed Computing*, 62(2):228–240, February 2002.

[14] I. Widjaja, A. Leon-Garcia, and H.T. Mouftah, The effect of cut-through switching on the performance of buffered banyan networks. *Computer Networks and ISDN Systems*, 26:139–159, 1993.

[15] 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.

[16] B.Y. Yu, Analysis of a dual-receiver node with high fault tolerance for ultrafast OTDM packet-switched shuffle networks. Technical paper, 3COM, 1998.

[17] B. Zhou and M. Atiquzzaman, Efficient analysis of multistage interconnection networks using finite output-buffered switching elements. *Computer Networks and ISDN Systems*, 28:1809–1829, 1996.

Postal addresses

Béla Almási
Institute of Informatics
University of Debrecen
H-4010 Debrecen
Egyetem tér 1.
Hungary

Dietmar Tutsch
Technische Universität Berlin
Einsteinufer 17
D-10587 Berlin
Germany