Analysis of Delay Time Distributions in Multistage Interconnection Networks Considering Multicast Traffic

Marcus Brenner and Armin Zimmermann
Technische Universität Berlin
Real-Time System and Robotics
D–10587 Berlin, Germany
{mbrenner, zimmermann}@cs.tu-berlin.de

Abstract

Multistage interconnection networks (Banyan networks) are proposed as connections in multiprocessor systems and in high-bandwidth network switches. In order to achieve suitable solutions when designing such networks to fit a given task, performance evaluation plays a crucial part. This paper presents an approximative analytical modeling approach that offers a performance measure (in addition to mean throughputs and delay times) that is considered important in real-time communications scenarios: distribution of delay times, for which no analytically tractable exact method exists.

1. Introduction

Multistage interconnection networks (MINs) are often proposed in high performance environments such as multiprocessor systems [1], high-bandwidth communication networks (e.g. ATM or Gigabit/10G Ethernet), and are candidates for application in distributed real-time systems.

MINs are used as well in embedded applications, providing the underlying interconnect between processing elements [11]. As the number of network nodes increases, their lower count of cross-points offers a considerable advantage over fully-meshed crossbars if cost or space constraints arise. With higher levels of integration becoming reality, multiprocessors on single chips are emerging. Several Network-On-Chip architectures for the interconnection network are the topic of current discussion [7]. Choosing a specific network design and topology is considered very application-dependant [5] and continues to be an active topic of research. Being able to evaluate the performance of such networks during the design phase is important as it allows to determine if a certain design fits the QoS requirements. Methods employed for performance evaluation include both analysis and simulation, each with its advantages and disadvantages. Specifically, there is a tradeoff between speed of execution of the analysis and accuracy of simulation. Also, [3] shows that obtaining quantile measures by simulation requires considerable computational effort compared to lower-order measures such as mean values. In the following, our focus will be on analytical performance evaluation and its use.

Applications of the proposed analysis method include design evaluation of Network-on-Chip topologies under multicast conditions or interconnection networks that provide links between processing elements in a large-scale multiprocessor environment. Being able to gauge performance measures including delay time distributions is an important property of a system development process.

In [4] Jenq developed an analytical model to cope with buffered MINs using just one input buffer to represent an entire network stage. Tutsch and Hommel [10, 8] extended Jenq's model to include packet multicasting and allowed for arbitrary (but finite) buffer sizes. They also considered dependence between packets of successive clock cycles. Their model yielded average throughputs and delays but no delay time distributions and thus was not suited to model real-time systems. To the best of the authors' knowledge, the proposed method is the only approach to obtain delay time distributions analytically.

This paper presents a model introduced in [2] and extends it to a networking application example, delivering delay time distributions while supporting multicast traffic. Delay time distribution analysis is inherently more complex than the computation of traditional expected (mean) values. Moreover, there are no numerically tractable exact methods. The presented method is able to cope with arbitrary network sizes, buffer capacities for the internal switching elements and arbitrary (but uniform) multicast traffic patterns. In order to validate the results obtainable with the analytical model, a state-based simulation was used.

978-0-7695-3192-2/08 $25.00 © 2008 IEEE
DOI 10.1109/NCA.2008.35

The remainder of this paper is organized as follows: Section 2 gives a short overview of MINs, Section 3 describes key ideas involved in developing the model. Section 4 presents an application of the proposed method to a 128 × 128 node network, focusing on the impact of multicast traffic on delay time distribution in particular. Finally, Section 5 gives a conclusion and points out directions for further studies.

2. Multistage Interconnection Networks

The $N \times N$-MINs considered in this paper (connecting N input ports to N output ports) consist of buffered 2×2 switching elements which are arranged in $n = \lceil \log_2 N \rceil$ stages. The MIN is internally clocked with all packet sending and receiving operations occurring simultaneously. Fig. 1 shows an 8×8-MIN to illustrate this structure.

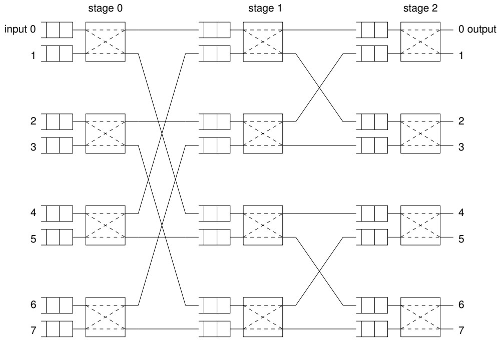

Figure 1. 8×8**-MIN**

Routing is performed according to a store-and-forward scheme, i.e., packets advance one network stage per clock cycle and are then stored in the stage's buffer until the next clock period.

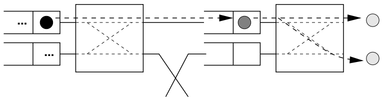

Figure 2. multicasting while routing

Packet multicasting can be handled by multistage interconnection networks in a bandwidth-conserving manner: packets that have multiple output ports as their destination are copied within the corresponding 2×2 switching element as required (Fig. 2, the packet is copied at the latest possible stage).

To be able to handle MINs by analytic means, one has to lower the complexity of the model. Here, this is accomplished by reducing each stage in the network to a single switching element. For this simplification to be valid, some prerequisites usually have to be assumed: the same input load is offered to all inputs, all packets have equal size, conflicts between packets are resolved randomly, and multicast traffic is uniformly distributed among the network outputs.

These conditions ensure that the buffers can be treated equally so that just one buffer can represent the behavior of all the buffers in the same network stage. Despite prohibiting analysis of non-uniform behavior occurring with hot-spot traffic or highly correlated data streams these assumptions can be considered valid for an average network usage scenario.

Leading to the model presented in the next section was the observation that deviation of delay measures from simulation results was considerably greater than this was the case for the throughput measures considered in [9].

3. Model and Analysis

Informally speaking, the discrete-time model describes one-step state transition probabilities. It is an approximation because not all individual model states are considered – similar ones are treated alike to save state space. Later on, the model is iteratively applied to an initial state probability vector, until convergence is reached (fix-point iteration). Performance measures of interest can then be derived from the resulting probability vector.

Due to space limitations, this section does only describe key aspects of the proposed model. Particularly, state transition equations are omitted and an overview of deriving these equations is presented instead.

As in [10], a stage of the MIN is described by two means: First, the type of packet in the first buffer position of the switching element and second, the number of packets waiting in the queue to be sent. All feasible combinations of packet types are considered states. The individual packet types used are: type *0* (empty buffer), type *n* (unicast packet), type *nb* (blocked unicast packet), type *b* (broadcast packet), type *bb1* (broadcast packet, one target buffer blocked), type *bb2* (broadcast packet, both target buffers blocked), and type *fb* (unicast packet, that is not in conflict with the unicast packet in the other buffer).

Based on these packet types the actual states of the model are determined. The states are composed of feasible combinations of packet types in the first positions of both buffers in a switching element. Considering these restrictions, the following 18 states (corresponding to their respective state probabilities π) have been identified to represent the first buffers of a switching element: $(0, 0)$, $(n, 0)$, $(nb, 0)$, $(b, 0)$, $(bb1, 0)$, $(bb2, 0)$, (n, n), (n, nb), (n, b), $(bb1, n)$, (nb, nb, c), (nb, nb, nc), $(bb1, nb)$, $(bb2, nb)$, (b, b), $(bb1, bb1)$, $(bb2, bb2)$ and (fb, fb). Two of these require additional explanation: In the case of two unicast packets, each with a blocked target buffer in the next network stage (states $(nb, nb, *)$), a differentiation is made regarding their destinations. If both packets have the same buffer as their target, the state (full or non-full) of the other target buffer is unknown. This state is designated (nb, nb, c). If these two packets do not compete for the same target (state (nb, nb, nc)), both their target buffers must be full.

Closely coupled with the actual states are the probabilities $r_*(k, t)$ for sending some or all packets in the switching element to the next stage. For every state exists a set of sending probabilities to describe all possibilities of packets leaving the switching element in that particular state.

In order to obtain delay time distributions, an additional quantity $l_m(k, t)$ is introduced that holds the probabilities of a switching element's buffer to contain a packet at position m (where 1 is the first buffer position) with a certain number of clock cycles expired: $l_m(k, t) = \left[l_{m,0}, l_{m,1}, \ldots, l_{m,\ell_{max}}\right]$

Initially (when the analysis starts with an empty network), this vector is set to zero for all m and k. New packets that enter the network have a zero delay time associated with them. Fig. 3 presents an example run for the first three clock cycles of an iteration under simplified conditions: 4 × 4-MIN, buffer size 1, offered input load 1, every multicast is a broadcast in each stage. At $t = 1$ two broadcast packets enter the first network stage, the second stage is still empty. In the next iteration step, one copy of each broadcast packet can be sent resulting in the state (nb, nb, nc) (probability 0.5) or one of the broadcast packets can be sent completely while the other remains in that stage (probability 0.5). Then, the buffer would be filled with a new broadcast packet immediately (because the probability of a packet being offered to the network inputs equals 1) which would turn into an blocked broadcast packet since both target buffers are full. In both cases, stage 1 contains two broadcast packets at $t = 2$.

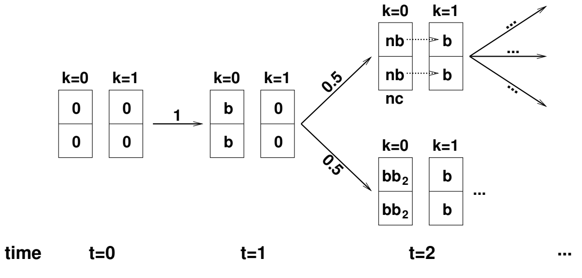

Figure 3. first iterations for maximum offered load and broadcasting for all packets

Using fixed-point iteration over the state probabilities, a steady state is reached from which the result parameters of interest are determined.

4. Results for an Application Example

The proposed method allows to efficiently determine the distribution of packet delay times in multistage interconnection networks (MINs) composed of 2 × 2 switching elements. In addition, mean throughputs and mean delay times are also determined as described in [2]. Input parameters of arbitrary choice for the analysis are the number of stages (thereby determining the network size), buffer size for each stage's switching element, the offered load and the multicast traffic pattern (when examining multicast traffic).

The following example assumes a 128 × 128 node network which is used as the underlying interconnect of a multiprocessor system. As it is considered an important ability to efficiently broadcast or multicast information during parallel and distributed computation tasks, a MIN is a viable choice for such an interconnect. Due to its property of performing broadcast at the appropriate stage during packet routing, bandwidth is used economically. For a characterization of the multicasts, it is assumed that tasks run localized on a small number of nodes as well as requiring information to be updated on many or almost all nodes.

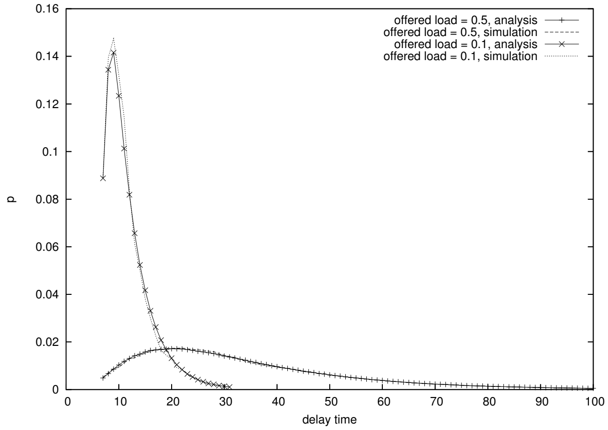

Figure 4. delay distribution: buffer size $m = 1$ with offered loads 0.5 and 0.1

Multicasting causes the network to saturate: Due to the multiplication of packets the buffers are almost always occupied at an offered load of 0.5. Only a small number of packets is able to traverse the network in just 7 clock cycles and the distribution of delay times becomes wider due to the uncertainty of the random packet-choosing process in the switching elements. Fig. 4 shows the corresponding delay distribution. This and the subsequent figure also compare the analysis' result to simulation, for that purpose the individual values have been connected by lines.

Offering a lower load (0.1), the corresponding delay time distribution shows a delay < 10 clock cycles for most packets with a rapid decline in probability towards higher delays. Considering the application's requirements, this could be an acceptable delay figure when specifying the maximum load the network will be able to handle within its performance specification.

In order to validate the proposed method, simulation has been used. This was done on packet level, measuring each delay time bin separately allowing for a 5% margin of error and a 95% confidence level. The highest deviation can be observed for packets that have been blocked multiple times because of network congestion. Compared to simulation, runtime of the analysis is considerably shorter: Obtaining delay-time figures by simulation typically takes several hours of computation time when adhering to the desired confidence levels and margins of error. Table 1 show runtime comparisons for MINs of different sizes.

network size	simulation runtime (sec)	analysis runtime (sec)
4 × 4	5	5
16 × 16	60	5
64 × 64	900	8
128 × 128	5400	10

Table 1. simulation vs. analysis runtime

5. Conclusion

The proposed model allows to analytically determine several important performance measures of multistage interconnection networks: mean throughputs at the inputs and outputs, mean delay times and additionally distributions of delay times. The latter measure is especially important when considering real-time applications such as audio/video transmission or other multimedia scenarios or network-on-chip communication architectures.

Due to the short amount of time required for the method to complete the analysis process, one can evaluate several sets of MIN parameters and thus improve network design to meet target specifications at low cost early in the design process.

Because of the model's state complexity, it does not appear viable to expand it to MINs that are composed of switching elements larger than 2 × 2. Because every combination of packet types in the first buffer position can (with the exception of infeasible states) become a model state, increasing the switching element size would cause an explosion of the state space. Simpler models do not to provide satisfactory results when one is interested in distributions of delay times and the network uses packet multicasting. This is because of the dependencies between adjacent buffers which are not considered by these models. These dependencies strongly affect the delay time performance measures and in particular the vector that is used to determine the distributions of delays.

Further studies will include characterization of the found delay time distributions from the analysis as well as allowing deadlines for packets. Then, discarding of packets that have exceeded their deadline and thus would be useless (e.g. in the context of a multimedia application like an audio/video transmission) would also be possible.

References

[1] G. A. Abandah and E. S. Davidson. Modeling the communication performance of the IBM SP2. In *Proc. of the 10th International Parallel Processing Symposium (IPPS'96); Hawaii*. IEEE Computer Society Press, 1996.

[2] M. Brenner. Improving accuracy in modeling multistage interconnection networks. In *Proc. of the High Performance Computing Symposium 2002; San Diego, USA*, pages 274–280. The Society for Modeling and Simulation International SCS, 2002.

[3] M. Eickhoff, D. McNickle, and K. Pawlikowski. Using parallel replications for sequential estimation of multiple steady state quantiles. In *Proc. of the 2nd Intl. Conf. on Performance Evaluation Methodologies and Tools (VALUETOOLS)*, 2007.

[4] 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, Dec. 1983.

[5] A. Jerraya, H. Tenhunen, and W. Wolf. Guest editors' introduction: Multiprocessor systems-on-chips. *IEEE Computer*, 7(38):36–40, 2005.

[6] J. K.-Y. Ng, S. Song, and B. Tang. A computation model for providing statistical performance guarantee to an ATM switch. *Real-Time Systems*, 23(3), November 2002.

[7] P. Pande, C. Grecu, A. Ivanov, R. Saleh, and G. De Michelli. Design, synthesis and test of networks on chips. *IEEE Design and Test of Computers*, 5(22):404–413, 2005.

[8] D. Tutsch and G. Hommel. Analysis of multicasting in buffered MINs. Tech. Report 1997–20, Technische Universität Berlin, 1997.

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

[10] D. Tutsch and G. Hommel. Multicasting in buffered multistage interconnection networks: an analytical algorithm. In *12th European Simulation Multiconference (ESM'98); Manchester*, pages 736–740. SCS, June 1998.

[11] W. Wolf. *High-Performance Embedded Computing*. Elsevier Inc., 2007.

[12] 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, Mar. 1990.