Performance Comparison of Wormhole-Routing Priority Switch Architectures

Michael Jurczyk
Department of Computer Engineering and Computer Science
University of Missouri-Columbia
Columbia, Missouri 65211, USA
mjurczyk@cecs.missouri.edu

Abstract – *Temporary nonuniform traffic patterns can severely degrade the performance of wormhole-routing multistage interconnection networks in multiprocessor systems. Temporary saturation trees build up inside the network under these traffic patterns that result in a temporary network overload with increased packet delay. Recently, enhanced switch box architectures and priority mechanisms were proposed that are able to alleviate or control the degrading effects of these saturation trees. In this paper, it is studied how these different mechanisms influence the lengths of hot-spot and overload phases, and message delay. All mechanisms are able to alleviate performance degradation. The switch box design proposed in [9] in conjunction with an alternating priority mechanism is able reduce network overload and message delay the most while the hot-spot phase is increased moderately only. This can be achieved with low hardware overhead.*

Keywords: multistage interconnection networks, network overload, priority switch architecture, saturation tree, wormhole-routing.

1. Introduction

In multiprocessor systems, multistage interconnection networks (MINs) are frequently used to interconnect the processors, or to connect the processors with memory modules, e.g., ASPRO, BBN Butterfly, Cedar, NEC Cenju-3, IBM RP3, IBM SP2, PASM, STARAN, Ultracomputer [13]. Blocking MINs consist of stages of switch boxes and provide a unique path between any source and destination pair, and different source/destination paths might share common links and/or switch boxes [8, 13]. In this paper, one class of MINs, the *multistage cube network* is considered, which is a representative of a family of topologies that includes the omega, indirect binary n-cube, delta, baseline, butterfly, and multistage shuffle-exchange networks [8]. In the discussions, wormhole-routing network organizations are assumed that connect the processors with the memory modules in a shared memory MIMD parallel computer. Results can also be applied to message passing systems.

Hot-spot traffic , in which many processors send data (hot messages) to the same destination, can cause congestion within the MIN and degrade the overall performance of the MIN substantially [3, 6]. If the traffic rate to the hot memory exceeds a certain threshold, a saturation tree of full switch buffers builds up from the last network stage and might even reach the source nodes. The network is overloaded and even messages not destined to the hot-spot destination are delayed substantially [3].

A variety of mechanisms can cause hot-spot traffic patterns in shared memory and distributed memory multiprocessor systems. For example, the access of a single shared variable by multiple processors can cause a hot-spot. Synchronization mechanisms (regular and Barrier Synchronization) where synchronization variables are used can produce severe hot-spots [3]. Also, algorithms such as Gaussian Elimination are prone to hot-spot contention [3]. Furthermore, cache coherence protocols can produce hot-spot patterns. The negative influence of those hot-spots on the overall network traffic has to be alleviated to obtain high performance in multiprocessor systems.

Several concepts to alleviate saturation tree effects on network performance have been proposed. These concepts can be divided into three classes: (1) combining techniques, (2) flow control techniques, and (3) enhanced switch box designs. In combining techniques, several messages destined to the same destination are combined into a single message to reduce the number of hot messages within the network [12]. This technique results in high hardware overhead, and might fail if hot-spot messages are not combinable due to multiple hot locations within a module. Also, combining techniques are unable to cope with traffic patterns that produce *nonuniform traffic spots* (*NUTS*) (e.g., the bit-reverse permutation traffic) that might cause congestion through multiple simultaneous partial saturation trees within a MIN. Flow control techniques, like feedback or discarding networks, often result in decreased performance under uniform traffic [4]. Enhanced switch box designs can alleviate saturation tree effects, while consuming less hardware, as compared to combining techniques, while they are also able to alleviate performance degradation under NUTS [9, 11].

This paper compares the performance of different enhanced switch box architectures and priority mechanisms under temporary nonuniform traffic patterns. It is shown how the different priority mechanisms are able to alleviate the network performance degradation due to temporary network overloads.

2. Network and Traffic Models

The networks considered in this paper are multistage cube networks [8] that connect N=2^n processors with N memory modules in a shared memory MIMD parallel computer. The networks are constructed from $s = \log_B N$ stages of $B\times B$ switch boxes. Each stage consists of N/B switch boxes; two consecutive stages are connected via N network links. A multistage cube network with N=8 and B=2 is shown in Figure 1.

Networks with a backpressure mechanism are assumed so that no messages are lost within the network. Also, a queue is associated with each processor that can buffer messages which cannot be injected into the network as a result of blocked network inputs (see Figure 1). This way, no messages are lost during a temporary network overload. Furthermore, wormhole-routing networks are assumed.

Wormhole-routing is a switching technique where a message is divided into several *flow-control digits* (*flits*) [10], which are the smallest unit of information transmitted between two switches within a network at once. The head of each message consists of one or more flits which contain the destination (routing) information, the last flit of each message contains an “End of Message” indication.

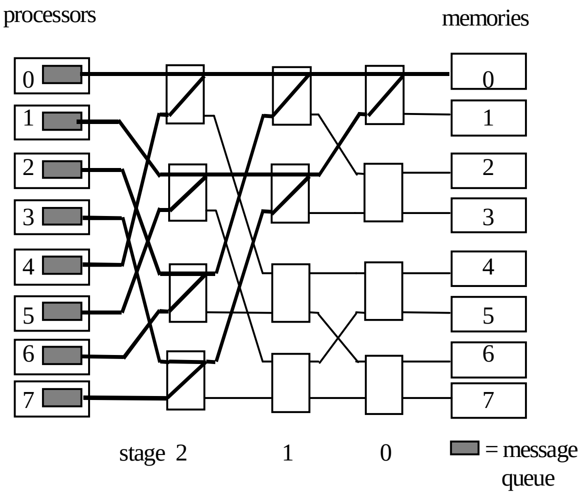


Figure 1: An 8×8 multistage cube network with 2×2 switch boxes; bold lines illustrate a saturation tree under hot-spot traffic to output 0

In Figure 2a), the architecture of a $B\times B$ wormhole routing switch box is exemplified by a 2×2 switch box. This architecture will be referred to in this paper as a *regular wormhole-routing switch box*. It consists of a 2×2 crossbar and a FIFO flit buffer (with a length of C flits) at each input to temporarily buffer flits that cannot be transferred through the switch due to busy output links or a full buffer in the next switch box. A simple handshake protocol between two switch boxes can be implemented with only a single bi-directional handshake line per switch box port [10]. During each network cycle, at most one flit can be transmitted over any network link.

For *uniform traffic*, each processor generates uniform messages with a fixed length of FU flits. The destinations of these messages are uniformly distributed. The traffic is characterized by the uniform traffic load λ ($0 \leq \lambda \leq 1$) that is defined as the number of uniform flits produced by a processor per network cycle.

If a set of processors accesses a single shared variable simultaneously, the processors belonging

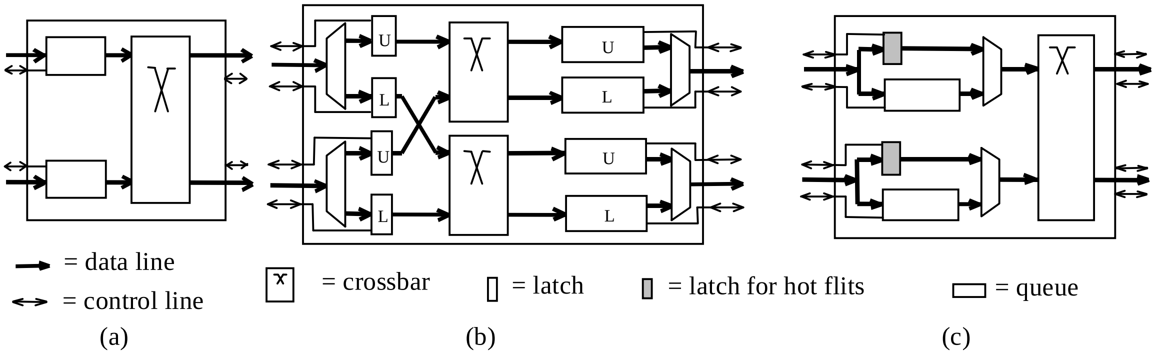


Figure 2: 2×2 (a) regular switch box, (b) switch proposed in [11], and (c) our priority switch [9]

to the set send a message (hot message) to the memory module the variable resides in, which results in a *temporary hot-spot traffic pattern* . Because the processors in a MIMD system are independent, they send their hot messages at different times. One way to model such hot message generation is a normal distribution of hot message generation over time with a mean μ and a standard deviation σ as proposed in [1]. It is assumed that each hot message has a fixed length of *FH* flits (*FH* < *FU*). Furthermore, it is assumed that while the hot-spot access is in progress, the processors perform a fast context switch and continue to work on a different program in the meantime. Thus, each processor generates uniform traffic with load λ before and after it emits a hot message. This kind of hot-spot scenario was chosen as one type of a worst-case dynamic traffic pattern in multistage interconnection networks by many researchers (e.g., [1, 9, 14]). The saturation tree that builds up under hot-spot traffic to memory 0 is depicted in Figure 1 by bold lines.

In many applications, *a priori* knowledge about occurrences of unsymmetrical data traffics is available, so that each processor of a parallel computer can distinguish different message classes [14], e.g., through the use of an intelligent compiler. Under hot-spot traffic scenarios, *hot messages* and *uniform messages* (messages belonging to the uniform background traffic), can be distinguished (and marked prior to entering the network). To determine the performance of the switch architectures under investigation, a third class of messages will be considered, the *uniform-hot messages* . These are messages that belong to the uniform traffic class but are destined to the hot-spot.

3. Temporary Hot-Spot Traffic in Networks with Regular Switches

1024×1024 multistage cube networks constructed from 2×2 regular wormhole-routing switch boxes (Figure 2a) are considered in this paper with a buffer capacity of 200 flits per input buffer (*C*=200). A temporary hot-spot traffic is assumed with λ=0.5, μ=4000, σ=50, *FU*=20, and *FH*=4. Extensive simulations show that results and conclusions drawn throughout this paper are valid for different network and switch box sizes and buffer and message lengths as well (the simulation results are not shown due to space limitation). The network was simulated with a parallel network simulator [7] running on a MasPar MP-1 SIMD computer with 16K nodes [13]. All results were averaged over 10 independent simulation runs.

Figure 3 illustrates the temporary hot-spot traffic effects. When no tree is present, the uniform message delay is approximately 160 cycles under that particular uniform traffic load. The temporarily filled saturation tree buffers during the hot-spot phase result in a delay increase of the uniform messages of up to 1,000 network cycles under that scenario. After most of the hot messages reached the hot destination (all *N* hot messages have traversed the network at cycle 8,900 as depicted in Figure 3a), the buffers in the saturation tree start to empty so that the average uniform message delay decreases again until the tree has completely vanished.

Considering Figure 3, two overlapping phases can be defined during temporary hot-spot traffic scenarios [5]: 1) the *hot-spot phase* , i.e., the time interval of length T_h from the injection of the first

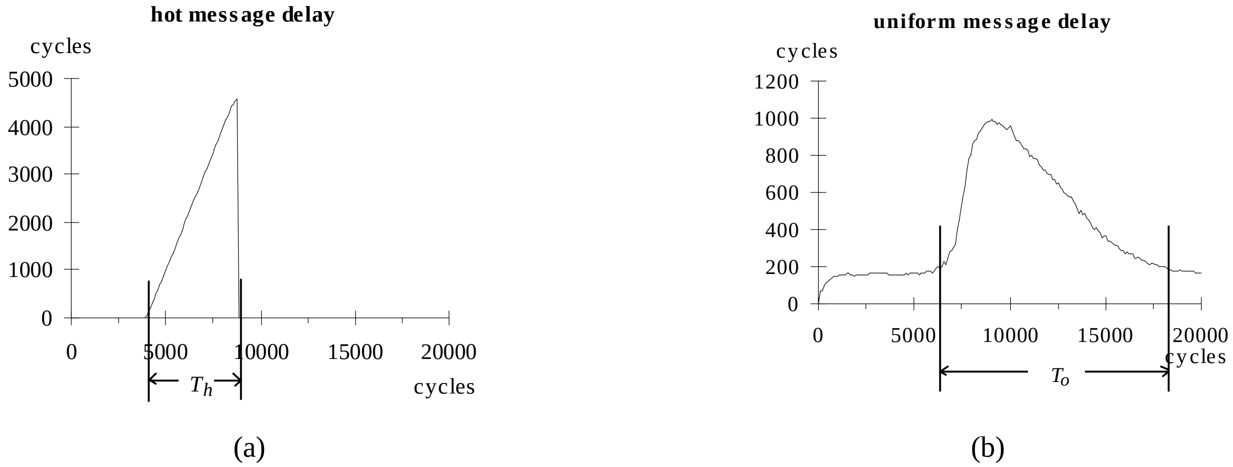


Figure 3: Delay of (a) hot and (b) uniform messages in a 1024×1024 network with 2×2 regular wormhole-routing switches with C=200 flits under hot-spot traffic with FU=20, FH=4, λ=0.5, μ=4000, and σ=50

hot message into the network until all hot messages left the network (see Figure 3a), and 2) the *overload phase*, i.e., the time interval of length T_o from the first detection of the saturation tree formation until the saturation tree has vanished (see Figure 3b).

4. Priority Switch Architectures

The switch architectures and priority mechanisms under investigation are introduced in this section.

A switch box architecture together with a priority scheme that alleviates the impact of the saturation tree on the uniform background traffic under hot-spot traffic was previously proposed in [11]. The architecture of that switch box is depicted in Figure 2b). There are two parallel input latches per input port and two queues per output port, forming two virtual channels [2] per switch port. A flit arriving at an input port is buffered in the U latch if it is destined to the upper switch box output port, while it is buffered in the L latch if it is destined to the lower switch box output port. Similarly, at an output port, a flit is buffered in the U queue if it is destined to the upper output port of the switch box in the next network stage, while it is buffered in the L queue if it is destined to the lower output port of the box in the next stage. To implement backpressure, two handshake lines per switch box port are needed. Compared to a regular switch box, this architecture has a duplicated internal data path that results in a considerable hardware overhead. However, because of the duplicated data path, saturation tree effects can be alleviated under hot-spot traffic scenarios, as was shown in [11]. While this architecture was proposed for packet switching networks, it can also be used for wormhole routing networks without any changes in the switch topology.

To increase the network performance under nonuniform traffic patterns even further, a priority scheme was proposed in [11] as well. In this priority scheme, messages destined to the hot network output are assigned a lower transfer priority as compared to all other messages buffered in an output queue. During a data transfer between stages, a data arbitration mechanism searches each output queue for the first message that is not destined to the hot-spot. This message is given priority over the messages buffered in front of it and is transferred to the next network stage. Under temporary hot-spot traffic, the messages not destined to the hot-spot are given transfer priority over all other messages within a queue, so that the uniform messages will leave a queue as early as possible. However, the hot messages will stay in the queues and will fill those eventually. A substantial hardware overhead is needed to implement this priority scheme because queues have to be searched and messages have to be processed in non-FIFO manner. To adapt their proposed priority scheme to the traffic scenario assumed here (hot messages are marked), mechanisms proposed in [11] to detect hot messages can be omitted here, resulting in an even better mechanism performance. The simulation study in [11] showed that the enhanced switch box architecture, together with the priority scheme can effectively reduce the network performance degradation under hot-spot traffic.

Another priority switch box architecture under investigation was proposed by the author in [9] and is depicted in Figure 2c). In this switch box architecture, a hot flit latch is employed in parallel to each uniform queue at each switch box input port, forming two virtual channels [2]. In the hot latches, only hot flits are stored. Uniform flits will be buffered in the uniform queues exclusively. Hot and uniform flits travel on the same inter-stage and crossbar links. During any network cycle, at most two flits are routed over the crossbar (at most one from the upper input port and at most one from the lower input port). It is assumed here that two separate bi-directional handshake lines per switch box port for hot and uniform flits are utilized (similar to the switch box proposed in [11]).

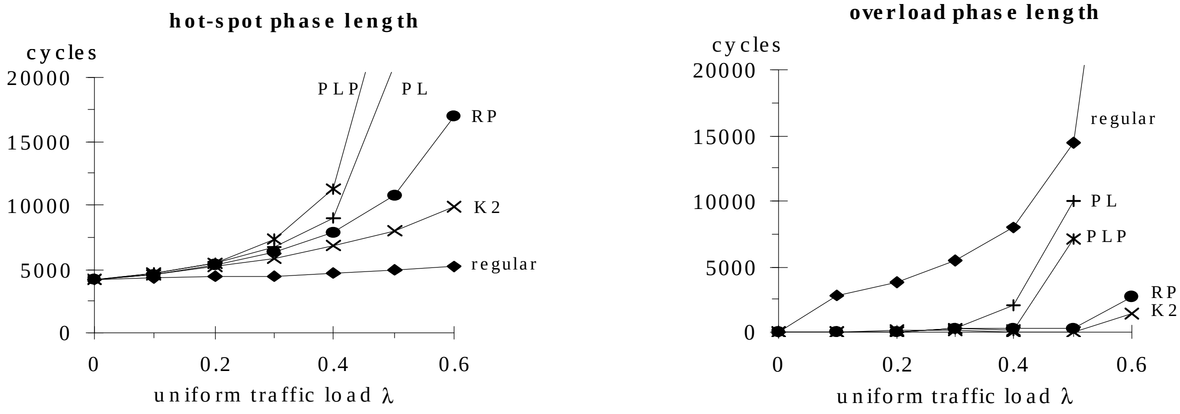
In this switch architecture, two buffers, i.e., the uniform queue and the hot latch, are competing for one crossbar input link (see Figure 2c). An *alternating transfer priority mechanism*, proposed in [9] determines in each network cycle, which buffer will transfer a flit over the crossbar link. Hot flits buffered in the hot latch at a switch box input port are given priority over the uniform flits in the related uniform queue only if at least K ($0 \le K < \infty$) uniform flits were transferred following the transfer of the last hot flit. If only flits in one of the queues are present, then those flits are transferred, independent of K. This scheme can be implemented by employing an additional K-counter at each switch box input port. In the case of K=0, hot flits have priority over uniform flits (a hot flit will be served immediately, while a uniform flit will only be transferred if no hot flits are present). If K is chosen to be very large, uniform flits have priority over hot flits (a uniform flit will be served immediately, while a hot flit will only be transferred if no uniform flits are present). In all other cases, the mechanism ensures that a certain fraction of the link bandwidth is reserved for hot and uniform flits respectively.

5. Performance Comparison

The performance of the following priority switch architectures and mechanisms under temporary hot-spot traffic scenarios are compared in this section: networks with (1) the switch proposed in [11] (Figure 2b) without their proposed priority mechanism (*PL*), (2) the switch proposed in [11] (Figure 2b) including their proposed priority mechanism (*PLP*), (3) a regular wormhole-routing switch (Figure 2a) (*regular*), (4) a regular wormhole-routing switch (Figure 2a) with Peir and Lee's priority scheme (*RP*), and (5) our priority switch (Figure 2c) with the priority parameter K=2 (*K2*). K=2 was chosen as a compromise between hot-spot phase and overload phase lengths; it results in a relatively short overload phase and a medium-length hot-spot phase [9]. To have a fair comparison, all switch architectures should have roughly the same hardware requirements. Therefore, the buffer size in the regular, the K2, and the RP switches is set to 200 flits, while the buffer size in the PL and PLP switches is set to 100 flits (because there are two parallel buffers per switch output port). However, recall that the PL and PLP switches still need a duplicated internal data path that is not present in the priority or regular switches.

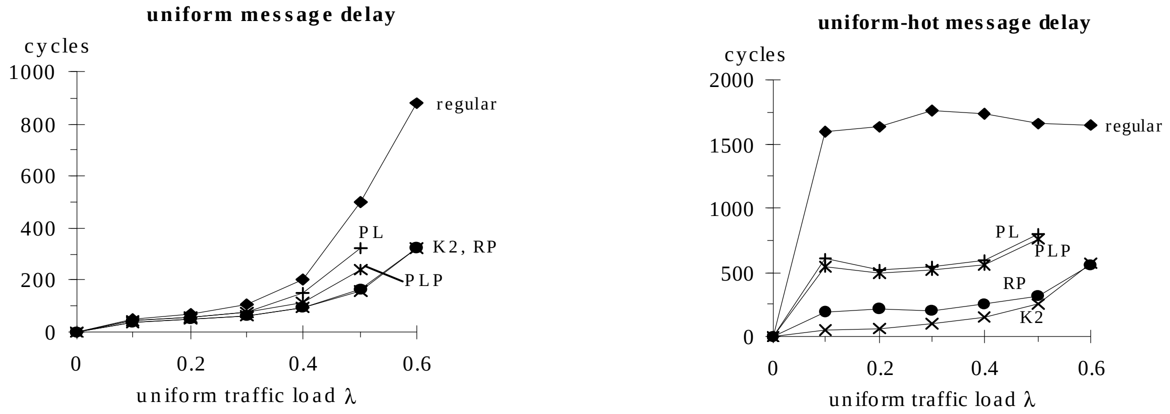
In Figures 4 and 5, the performance of the different networks is shown. All four enhanced designs are able to alleviate the performance degradation (lower message delay and shorter overload phase) as compared to the regular switch box case. This is traded off with a longer hot-spot phase length. The PL and PLP networks work only for loads of $\lambda \le 0.5$. At higher loads the network becomes overloaded even under pure uniform traffic. This is due to the shorter buffers in these switches. Thus, when employing wormhole-routing, the switch box architecture proposed in [11] might result in a degraded performance under uniform traffic which is not desirable (to avoid the performance degradation, larger buffers have to be used within the switch). Nevertheless, the PL and PLP networks are able to alleviate the performance degradation under nonuniform traffic substantially.

Our K2 network works in the full traffic load range depicted and performs the best (shortest overload phase, and lowest message delays), while the length of the hot-spot phase is only moderately increased as compared to all other networks. The RP network can also effectively alleviate the performance degradation. It performs almost as good as our K2 network when the overload phase length, and the message delays are considered. However, in the PL, PLP, and RP networks, the hot-spot phase is much



(a) (b)

Figure 4: (a) Hot-spot phase and (b) overload phase length in a1024×1024 network with 2×2 regular, PL, PLP, K2, and RP switches with C=200 flits under hot-spot traffic with FU=20, FH=4, μ=4000, and σ=50



(a) (b)

Figure 5: Overload phase delay of (a) uniform and (b) uniform-hot messages in a 1024×1024 network with 2×2 regular, PL, PLP, K2, and RP switches with C=200 flits under hot-spot traffic with FU=20, FH=4, μ=4000, and σ=50

longer than in our K2 network, especially for higher traffic loads. The priority mechanism proposed in [11] implemented in regular wormhole-routing networks is quite effective in alleviating performance degradation due to temporary saturation trees. However, a substantial hardware overhead is needed to implement this priority scheme. In contrast, our K2 switch architecture is able to further alleviate the performance degradation while consuming less hardware. Furthermore, one has to keep in mind that the performance behavior cannot be changed in the PL, PLP, and RP networks, but can be controlled in our priority network by changing the parameter *K*. For example, the overload phase length at λ=0.6 can be further reduced by increasing *K* that will still result in a smaller hot-spot phase length than in the PL, PLP, or RP networks [9].

6. Conclusion

This paper compared the performance of different enhanced switch box architectures and priority mechanisms under temporary nonuniform traffic patterns. An enhanced switch architecture and a priority mechanism previously proposed in the literature was compared to a switch architecture recently proposed by the author. It was shown that the previously proposed switch architecture suffers from the fact that a duplicated internal data path is needed, resulting in smaller buffer sizes as compared to the other switch designs (with comparable hardware requirements). It was also shown that the previously proposed priority mechanism implemented in regular wormhole-routing switches is quite effective in alleviating the performance degradation. It was furthermore shown that our architecture is most effective in alleviating the performance degradation on the uniform background traffic and the overload phase under temporary hot-spot traffic scenarios, while the hot-spot phase is only moderately increased as compared to all other mechanisms under investigation. This architecture is able to reduce the network overload phase length by up to 97%, and uniform and uniform-hot message delays by up to 63%.

References

[1] S. Abraham and K. Padmanabhan, "Performance of the direct binary n-cube network for multiprocessors," *IEEE Transactions on Computers*, Vol. C-38, No. 7, July 1989, pp. 1000-1011.

[2] W. J. Dally, "Virtual-channel flow control," *IEEE Transactions on Parallel and Distributed Systems*, Vol. 3, No. 2, March 1992, pp. 194-205.

[3] S. P. Dandamudi, "Reducing hot-spot contention in shared memory multiprocessor systems," *IEEE Concurrency*, Vol. 5, No. 1, Jan.-March 1999, pp. 48-59.

[4] W.S. Ho and D.L. Eager, "A novel strategy for controlling hot spot congestion," *1989 International Conference on Parallel Processing*, August 1989, pp. 14-18.

[5] M. Jurczyk and T. Schwederski, "Switch box architecture for saturation tree effect minimization in multistage interconnection networks," *1995 International Conference on Parallel Processing*, August 1995, pp. I/41-I/45.

[6] M. Jurczyk and T. Schwederski, "Phenomenon of higher order head-of-line blocking in multistage interconnection networks under nonuniform traffic patterns," *IEICE Transactions on Information and Systems*, Vol. E79-D, No. 8, August 1996, pp. 1124-1129.

[7] M. Jurczyk, T. Schwederski, H. J. Siegel, S. Abraham, and R. Born, "Strategies for the implementation of interconnection network simulators on parallel computers," *International Journal of Computer Systems Science & Engineering*, Vol. 13, No. 1, January 1998, pp. 5-16.

[8] M. Jurczyk, H. J. Siegel, and C. Stunkel, "Interconnection Networks for Parallel Computers" in *Encyclopedia of Electrical and Electronics Engineering* , J. G. Webster, ed., John Wiley and Sons, New York, NY, 1999, pp. 555-564.

[9] M. Jurczyk, "Traffic control in wormhole routing multistage interconnection networks," *IASTED International Conference on Parallel and Distributed Computing and Systems*, Vol. 1, November 2000, pp. 157-162.

[10] L. M. Ni and P. K. McKinley, "A survey of wormhole routing techniques in direct networks," *IEEE Computer*, Vol. 26, No. 2, February 1993, pp. 62-76.

[11] J.-K. Peir and Y.-H. Lee, "Look-ahead routing switches for multistage interconnection networks," *Journal of Parallel and Distributed Computing*, Vol. 19, No. 1, September 1993, pp. 1-10.

[12] G. F. Pfister and V. A. Norton, "`Hot spot' contention and combining in multistage interconnection networks," *IEEE Transactions on Computers*, Vol. C-34, No. 10, October 1985, pp. 933-938.

[13] T. Schwederski and M. Jurczyk, *Interconnection Networks: Structures and Properties (in German)*, Teubner Verlag, Stuttgart, Germany, 1996.

[14] M.-C. Wang, H. J. Siegel, M. A. Nichols, and S. Abraham, "Using a multipath network for reducing the effect of hot spots," *IEEE Transactions on Parallel and Distributed Systems*, Vol. 6, No. 3, March 1995, pp. 252-268.