Interconnection Networks II

Bidirectional Ring: An Alternative to the Hierarchy of Unidirectional Rings

Muhammad Jaseemuddin and Zvonko G. Vranesic

Department of Electrical & Computer Engineering
University of Toronto, Toronto, Canada, M5S 1A4
Email: (jaseemud, zvonko)@eecg.toronto.edu
Tel: 416-978-5032, Fax: 416-971-2326

Abstract. A hierarchy of unidirectional rings has been used successfully in distributed shared-memory multiprocessors. The fixed cluster size of the hierarchy prevents full exploitation of communication locality. The bidirectional ring is presented as an alternative to the hierarchy. Its relative performance is evaluated for a variety of memory access patterns and network sizes. It gives superior performance for low communication locality and for large networks. Another useful feature of the bidirectional ring is that the network load tends to be balanced over the two constituent unidirectional rings. These features make the bidirectional ring an attractive possibility as a network structure for scalable NUMA multiprocessors.

1 Introduction

A hierarchy of unidirectional bit-parallel rings has been used successfully in distributed shared-memory multiprocessors [2, 8]. In a hierarchical design, clusters of processors are connected in a fashion that enables addition of more clusters as the machine expands. Clustering is an architectural feature that is effective in exploiting communication locality of parallel programs. Unidirectional rings have been used to form clusters of processors, as well as to interconnect clusters in the hierarchy. The cluster size is dependent on the configuration of a particular machine. This may hinder full exploitation of locality when an application does not map well onto the clusters that are naturally available.

This paper proposes a bidirectional ring that exhibits dynamic clustering and exploits communication locality very effectively. In this case, a cluster is formed by spanning any number of consecutive nodes in the ring. This is done simply by allocating the required number of contiguous processors to a process that is conducive to a cluster of a particular size.

The paper gives an assessment of the relative performance of the bidirectional ring in comparison with the hierarchy of unidirectional rings. The problems of static clustering of the hierarchy are discussed in section 2. Section 3 introduces the bidirectional ring and dynamic clustering. The simulation environment is described in section 4, and the results of simulations are presented in section 5. Finally, the main results are summarized in section 6.

2 Hierarchy of Unidirectional Rings

The hierarchy of unidirectional rings that we will use for comparison purposes is similar to the hierarchy used in Hector [8] and KSR1 [2] multiprocessors. Figure 1*a* shows a simple example of a 2-level hierarchy, in which *local* rings connect a certain number of nodes and these rings are interconnected by a second-level ring. A node consists of a processor, a memory unit (holding a portion of shared address-space), and a communication switch. Larger hierarchies are implemented by increasing the number of levels in the system.

Each local ring in the hierarchy forms a cluster of processors. Thus, the size of a cluster is fixed. The static nature of the cluster causes two problems. First, it favors the assignment of processors to a process to suit the cluster size. Second, the performance of the network becomes very sensitive to the memory access patterns that exhibit variable amount of inter-cluster communication. A process that spawns as many threads as the number of processors in a cluster can naturally fit in one cluster, but the addition of one more thread causes it to span two local rings. This introduces a communication overhead proportional to the size of the second-level ring connecting the two local rings. We call this the *spanning overhead.* If it takes more levels for a packet to cross to reach the destination ring, then each level will add a factor to the overhead proportional to the size of the ring at that level. Furthermore, when a packet ascends or descends the hierarchy it passes through several inter-ring interface buffers, each of which contributes a variable amount of time to the overhead. In this case, depending upon the system load and the amount of inter-cluster communication, the benefits of more parallelism (threads) may diminish.

The hierarchy is particularly effective in exploiting the communication locality that is concentrated within clusters. It is less effective when there exists a low degree of communication locality where each processor communicates frequently with processors that are not necessarily a part of the same cluster.

3 Bidirectional Ring

Bidirectional rings have been considered mainly for the local area network environment [3]. In this paper, the bidirectional ring is viewed as a network structure for a multiprocessor, where due to communication locality the proximity of interacting processors is the main issue to deal with. A key property of the bidirectional ring is dynamic clustering.

In the bidirectional ring data traffic flows in both clockwise and counter clockwise directions. This is realized by having two rings that run in opposite directions, as shown in figure 1*b*. A node selects the ring to transmit to another node based on the shortest distance between the source and the destination. This keeps the distance between the two nodes to not more than half the ring size. A simple transmission scheme based on a request-response protocol, discussed in section 4.2, can be used to control the flow of traffic on the rings.

The bidirectional ring adapts to the application and forms clusters of processors dynamically. If a process is scheduled to run on a set of adjacent processors,

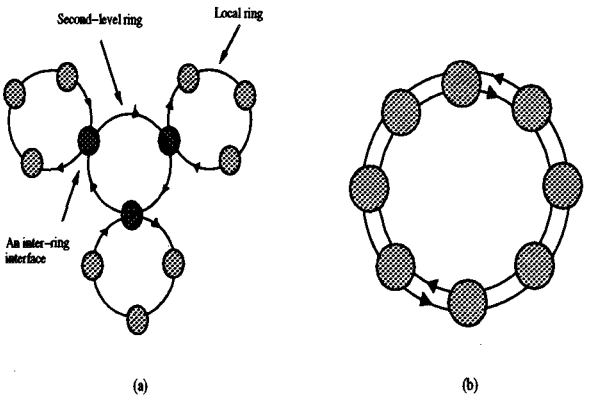

Fig. 1. An example of: (a) the hierarchy of unidirectional rings, and (b) bidirectional ring

then these processors automatically form a cluster where all communication among the processors takes place within a portion of the ring occupied by this cluster. Thus, traffic created by two or more processes tends to be contained within their own clusters. The number of processors forming a cluster is not constrained by the physical structure of the network.

The dynamic clustering in a bidirectional ring offers two advantages over the hierarchy of unidirectional rings. First, the spanning overhead is non-existent, which improves the performance of a process scheduled to run on two or more clusters (local rings) in the hierarchy. However, expanding the cluster increases the communication cost proportional to the additional links. Second, the variable size of the cluster promotes assigning a processor to every thread. This allows the number of threads to match the desired parallelism (the parallelism exhibited by the algorithm) of a large program.

Clusters in a bidirectional ring are not rings in the true sense; rather, they form chain-like structures in both directions. The round trip delay varies depending upon the positions of the source and the destination in the cluster. The maximum round trip delay is of the order of $2P$, where P is the number of nodes in the cluster.

The bidirectional ring is expected to show a smooth performance improvement with increasing locality. This is because, unlike the hierarchy, the bidirectional ring does not show jumps in the communication cost. Instead, it increases proportionally as the distance between the nodes is increased. Therefore, for each node, the cost of communicating with a node of equivalent distance is the same. As the communication locality increases, the neighborhood of a given node decreases; that is, the node communicates mostly with only a few of its nearest neighbors.

4 Experimental Setup

We have simulated the bidirectional ring and the hierarchy of unidirectional rings of different sizes to assess their relative performance using a simple but real communication protocol and synthetic workloads that are tuned by carefully selecting the parameters to reflect the actual workloads. A similar approach is used in evaluating the performance of a hierarchical ring-based system in [5]. The use of synthetic workloads allows us to study the behavior of each network under many possible interesting situations, and it also makes the simulation of large-scale networks manageable. In [1] real applications are used to analyze medium scale systems using a slotted-ring and running different cache coherence protocols. A different approach is used in [7], where first an analytical model of the SCI ring is developed and then a simple synthetic workload is used to analyze and validate the analytical model with the simulation of the actual network. This section explains the environment and parameters of the simulator. Section 4.1 describes the system model. The network protocol is discussed in section 4.2. Section 4.3 explains different workload models used in the simulation.

4.1 System Model

A node consists of a processor, a memory unit and a communication switch. The network is connected to the memory through the switch, and receives only those memory requests which are not satisfied by the local memory. This is modeled by taking a fraction of all requests generated by the processor. The request rate is a parameter which indicates the average number of requests generated by the processor per cycle. It is fixed at 0.05 which corresponds to 20 cycles between two consecutive cache misses. This rate is supported by real workloads [5]. The ring cycle time and the processor clock cycle time are considered to be the same. The ratio of the ring cycle time to the memory access time is 10:1. For instance, a word read takes 10 ring cycles. However, for a cache line read, the first word takes 10 cycles, but each subsequent word is available after every 5 cycles. The ring is a slotted ring [6].

The communication locality can be modeled as groups of *nearby* nodes around a source processor and assigning access probability to each group indicating that the target memory is in that group when the request is not to the local memory. This model is similar to the clusters of communication locality model used in [5]. A group of size s and access probability p consists of all nodes within a distance of $s/2$ from the source. A request generated at the source goes to a node in this group with a probability p given that the request is not to the local memory. The distribution of destination addresses within a group is uniform. For instance, in a system of 1024 processors with a local memory target probability of 0.9, $GS = (512, 768, 1024)$ and $GP = (0.8, 0.95, 1.0)$ defines 3 groups, where GS and GP indicate group sizes and their corresponding access probabilities. The local memory serves on average 90% of all requests. The first group consists of 512 nodes and 80% of the remaining remote requests involve nodes within this group. The second group is formed by adding 256 more nodes to the 512 nodes

of the first group. The access probability of this enlarged group is 0.95, meaning that 95% of remote requests go to the nodes of this group. Finally, all 1024 nodes form the third group, and all remote requests are within this group. The communication locality defines the local memory target probability, the number of groups, the size of each group, and the access probability of each group.

The invalidation packets introduced by the cache coherence scheme are not considered in the network traffic. Ignoring the invalidation traffic has no major effect on the overall results. For common applications, average number of shared writes causing invalidations is not more than 2 to 3 percent of total requests, and the average invalidation per shared write involves not more than 2 packets [4]. Assuming one packet per invalidation, the percentage of invalidation packets becomes less than 10% of the total request packets. This value decreases further if it is considered as a fraction of the sum of request and response packets.

4.2 Request-Response Protocol

A Hector-like request-response protocol [8] is used for network transactions. A subset of possible transactions is used which is sufficient to give a good indication of the relative performance that may be expected from the bidirectional ring. The three types of requests modeled are: i) READ a memory word, ii) READ a CACHE LINE (Cache line transfer), and iii) WRITE a memory word. Each processor can have only one outstanding request. A single packet is used to transmit a request. The response can involve multiple packets, such as for a READ CACHE LINE. The data to be written is included in the WRITE packet. For writes, the acknowledgments are issued after queuing the WRITE packets for later storing in the memory. A negative acknowledgement (NACK) is issued when the request cannot be buffered at the receiving node. The sender retransmits the same request after receiving the NACK. Another occasion when a sender retries is after a time-out, which is a mechanism used to recover from a packet loss. When a node cannot gain access to the network, it is said to be *blocked*, and the time it remains blocked is called *blocking time*. During this period the node continues to serve requests from remote processors.

In the hierarchy of unidirectional rings a WRITE packet is turned into an acknowledgement packet by simply changing the request type. In a bidirectional ring this scheme cannot work, otherwise the advantage of shorter distance will be compromised. A separate acknowledgement packet is formed and transmitted on the ring running in the opposite direction. This requires an acknowledgement buffer for each ring at every node to queue the acknowledgement packets for later transmission when the network is busy. We used this simple protocol to highlight the potential problem spots in the network structures. We used a 32-deep receiving and acknowledgement buffer each, and a 64-deep inter-ring interface buffer in the simulated system.

4.3 Workload Models

A number of workloads have been used to model different request traffic patterns. It is shown in [5] that substantial ring contention is observed when more than 10% of memory requests are to remote memory modules. Hence, to examine the network's effect on latency, 80% of requests are targeted to the local memory, and 20% are transmitted through the network to remote memories. The salient features of the workload models are discussed below.

1. **Bursty:** This workload models multiple request packets transmitted in a bunch, such as for cache line writes, multiple outstanding requests, and cache line prefetching. The two parameters that characterize this model are the average number of packets in a burst (burst length), and the average interval between two bursts. The average burst length is set to 5, and the average interval is fixed at 100. All the packets in a burst are directed to the same destination. This is because: i) a cache line write is sent to a single destination, and ii) due to spatial locality consecutive cache misses are served by the same target memory.
2. **Uniform:** This workload model selects the destination node randomly within a group.
3. **Hotspot:** There are periods when requests originating at different processors are destined to the same memory module due to synchronization or false sharing. In such a situation a single node receives a slew of requests from many nodes. Different hotspots are selected randomly each for 10% of total hotspot transactions. Hotspot transactions are 3% of total requests, which is in the range of practical values [5]. The other 97% of packets are generated using the uniform workload model.
4. **Mirror:** The destination address is predetermined for each source and is at the same distance from the other extreme as the source is from one extreme, that is $destination = (number\ of\ nodes - source)\ mod\ number\ of\ nodes$. This model does not exhibit communication locality, thus it serves as the stress test for a network designed to exploit communication locality.

5 Comparing Performance of the Bidirectional Ring and the Hierarchy of Unidirectional Rings

The main performance metric used to compare the performance of different networks is the request latency, which is defined as the number of cycles spent from the time when a processor issues a request to the memory to the time a response is received from the memory. Thus, the request latency is a sum of blocking times at both ends (source and destination), network propagation delays, and cumulative time spent in all the inter-ring interface buffers by the request and the response packets. Different memory requests experience different latencies. The *average latency* for each request type is computed first, and then the weighted averages of the three types of latencies are calculated.

Another important performance measure is *network utilization*, which is defined as the average percentage of busy slots over the total number of slots in the ring at any given time. This average is calculated over the total number of execution cycles. For the results presented in this section, a single application is assumed to be running.

Systems of various sizes, ranging from 16 to 1024 processors, have been simulated using the above workload models. Following are the three different network configurations considered.

1. **Hierarchy:** This is a hierarchy of unidirectional rings where 16 processors are attached to each of the lowest-level rings. Table 1 shows the configuration of different sizes in terms of the branching factor of each level of the hierarchy, from the lowest to the highest level.

Table 1. Configurations of different sizes

No. of Processors	Hierarchy
16	16,1
32	16,2
64	16,4
128	16,2,2,2
256	16,4,2,2
512	16,4,4,2
1024	16,4,4,4

2. **Bi-1:** A bidirectional ring where the channel width of each of the two rings is half the channel width of a unidirectional ring in the hierarchy. This is simulated by doubling the number of packets per cache line transfer on the bidirectional ring in comparison with the hierarchy. A request is assumed to be transmitted in a single packet.
3. **Bi-2:** A bidirectional ring where each ring has the same channel width as the hierarchy.

Two models of communication locality are considered, where the system consisting of N nodes is divided into three groups of communication locality:

1. **CLM1:** Local memory target probability=0.8, $GS = \{N/2, 3N/4, N\}$, and access probability is assumed to be $GP = \{0.8, 0.95, 1.0\}$. This represents an application showing low communication locality.
2. **CLM2:** Local memory target probability=0.8, $GS = \{4, 20, N\}$, and the access probability is assumed to be $GP = \{0.8, 0.95, 1.0\}$. This is representative of applications showing very high communication locality.

5.1 Network Performance

In this section, the simulation results for three workload models—uniform, bursty, and mirror—are presented. Hotspot workload is discussed in the next section. For brevity, the term hierarchy is used for the hierarchy of unidirectional rings, and latency for the average latency.

Uniform Workload. For small networks, the performance of the hierarchy and Bi-1 is not significantly different, as shown in figure 2. Latency in the hierarchy becomes considerably worse for large networks, mainly because of congestion at high level rings. In this situation inter-ring interfaces become points of high activity, even losing some packets, as shown in figure 3. It is interesting to observe the effects of high utilization of the high-level ring in the hierarchy in comparison with the bidirectional ring. In the hierarchy, inter-ring interface buffers are congested resulting in some packet losses. On the other hand, high utilization of the constituent rings in a bidirectional ring shows high contention for the network, but does not result in loss of packets, as indicated by increasing blocking time in figure 3.

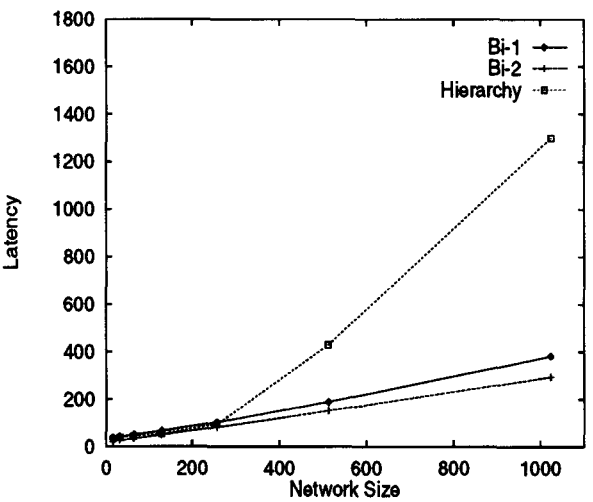

Fig. 2. Latency for uniform traffic and CLM1

The simulation results show a marginal difference between the utilization of the two rings in a bidirectional ring, which is due to the fact that different rings are used for transfer of request and response packets between any two nodes. Hence, the network load tends to be balanced between the two rings.

The causes of packet loss in the two networks are different. In the hierarchy, packets are lost mainly due to the lack of buffer space at the inter-ring interfaces. In a bidirectional ring packets are lost when acknowledgement packets are

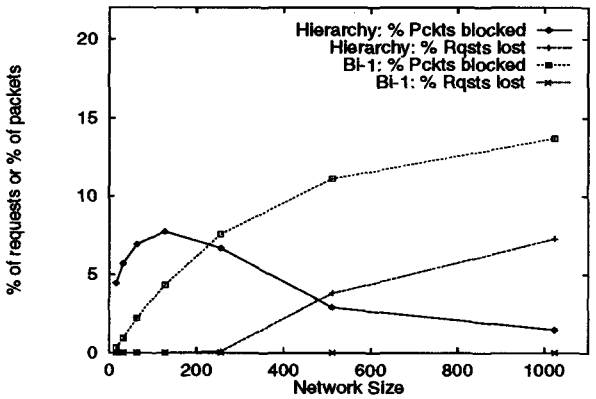

Fig. 3. Blocked Packets and Lost Requests for uniform traffic and CLM1

dropped due to the shortage of acknowledgement buffer space. This occurs very seldom, and appears only in those cases where momentary bursts of requests are received by some nodes. Therefore, packet loss is only observed in the hotspot workload.

Bi-2 results in improved performance for networks of all sizes, as shown in figure 2.

The hierarchy is very sensitive to communication locality. In CLM2, 96% of communication takes place within a distance of 4; that is, most of the communication is confined to the local ring. As shown in figure 4, the hierarchy shows a large reduction in latency for CLM2, as compared to CLM1. The response of a bidirectional ring is also favorable to high communication locality, and its performance is comparable to the performance of the hierarchy.

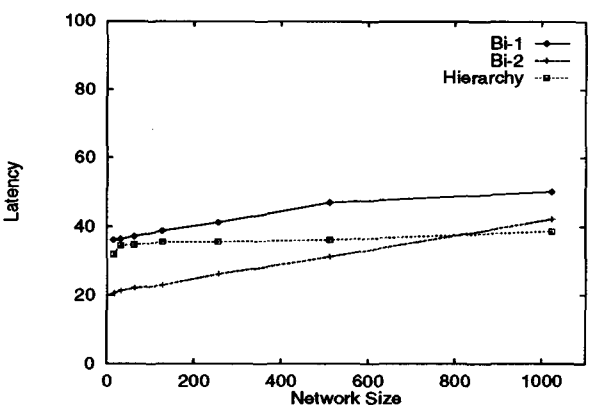

Fig. 4. Latency for uniform traffic and CLM2

Bursty and Mirror Workloads. Results for the bursty workload are shown in figure 5. The latency of the hierarchy for small systems is slightly less than the latency of Bi-1. For large systems the hierarchy shows much worse latency.

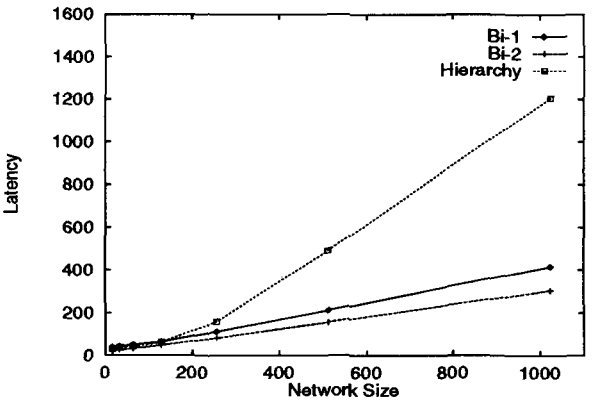

Fig. 5. Latency for bursty traffic and CLM1

Lack of communication locality in the mirror workload increases the latency in both types of networks, as shown in figure 6. The adverse effect is more pronounced in the hierarchy, because a large number of transactions occur between the nodes located in different rings which causes packets to traverse many levels. Every time a packet moves either up or down in the hierarchy, it may spend some time in an inter-ring interface buffer.

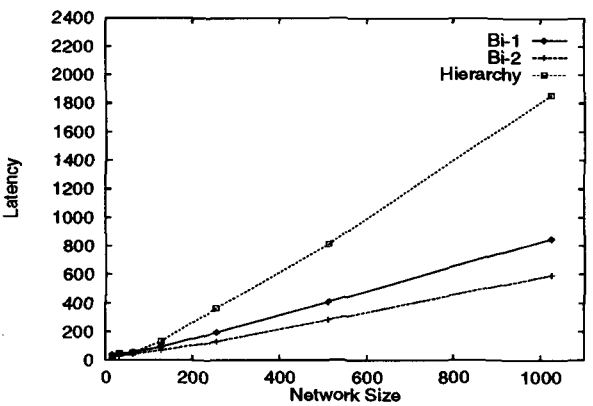

Fig. 6. Latency for mirror traffic and CLM1

5.2 Hotspot Workload

The relative performance of the bidirectional rings and the hierarchy for hotspot workload shows the same trends found for other workload models, as indicated in figure 7. However, the difference between the latencies of the hierarchy and Bi-1 is much less for large networks, as compared to the uniform workload case. The main reason for the smaller difference is the large number of retries that appear on the bidirectional ring. In the hierarchy, a request packet passes through several buffers, one at each inter-ring interface, before reaching the destination node. This spreads out the load at the buffer of the destination node. In contrast, a single buffer in the bidirectional ring receives an equivalent number of requests. This results in many requests being dropped after reaching the destination, which in the hierarchy may find space in one of the inter-ring interface buffers. Another important difference is the contribution of a retry to the latencies on the two networks. In the hierarchy a request may be dropped by an inter-ring interface, which adds less time to the latency than would be the case if the request is dropped by the destination node. In contrast, a request may be dropped only after reaching the destination in the bidirectional ring, and thus every retry adds a round trip delay to the latency.

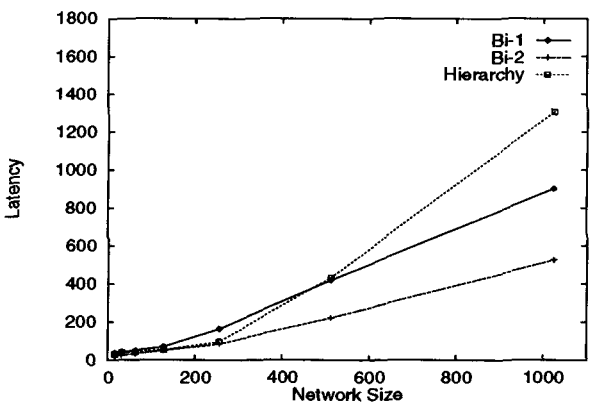

Fig. 7. Latency for hotspot traffic and CLM1

6 Concluding Remarks

The bidirectional ring is presented as an alternative to the hierarchy of unidirectional rings. The relative performance of these networks has been evaluated for different memory access patterns and network sizes, ranging from 16 to 1K processors. The fixed cluster size of the hierarchy makes it very sensitive to the communication locality, especially for large networks. In contrast, the bidirectional ring shows a smooth improvement in performance as the communication

locality is increased. The optimal locality for the hierarchy occurs when the communication is confined within clusters. Even for such workloads the performance of the bidirectional ring is comparable to the hierarchy. For low communication locality the bidirectional ring outperforms the hierarchy for all network sizes. In the absence of communication locality, the degradation in the performance of the bidirectional ring is much less pronounced than in the hierarchy.

For hotspot workload, the bidirectional ring outperforms the hierarchy, but not as significantly as in the case of other workload models. In the bidirectional ring, the receiver buffer at the hotspot creates a problem in handling momentary bursts of requests. This is less of a problem in the hierarchy, because the inter-ring interface buffers in the communication path share the load.

Another advantage of the bidirectional ring over the hierarchy is the load balancing in the network. Under certain conditions, such as low communication locality, the hierarchy may experience problems with saturation of high-level rings, whereas the low-level rings remain underutilized at the same time. In contrast, the bidirectional ring shows an effectively uniform load on both rings.

The third advantage of the bidirectional ring is that it has better reliability because a communication protocol may be defined that can tolerate the failure of one link. These factors make the bidirectional ring an attractive choice for scalable NUMA multiprocessors.

References

1. L.A. Barroso, and M. Dubois, *The Performance of Cache-Coherent Ring-based Multiprocessors*, Proc. ISCA, pp. 268-277, May 1993.
2. H. Burkhardt III et al, *Overview of the KSR1 Computer System*, Technical Report KSR-TR 9202001, Kendall Square research, February 1992.
3. I. Cidon, and Yoram Ofek, *MetaRing - A Full-Duplex Ring with Fairness and Spatial Reuse*, IEEE Trans. Communications, Vol. 41, No. 1, pp. 110-120, January 1993.
4. A. Gupta, and W.-D. Weber, *Cache Invalidation Patterns in Shared-Memory Multiprocessors*, IEEE Trans. Computer, Vol. 41, No. 7, pp. 794-810, July 1992.
5. M. Holliday, and M. Stumm, *Performance Evaluation of Hierarchical Ring-Based Shared Memory Multiprocessors*, IEEE Trans. Computer, Vol. 42, No. 1, pp. 52-67, January 1994.
6. A. Hopper, and R. Williamson, *Design and Use of an Integrated Cambridge Ring*, IEEE Journal on Selected Areas in Communications, November 1983.
7. S.L. Scott, J.R. Goodman, and M.K. Vernon, *Performance of the SCI Ring*, Proc. ISCA, pp. 403-412, May 1992.
8. Z.G. Vranesic, M. Stumm, D.M. Lewis, and R. White, *Hector - a Hierarchically Structured Shared-Memory Multiprocessor*, IEEE Computer, Vol. 24, No. 1, pp. 72-79, January 1991.