# Sliding-banyan network performance analysis

Michael W. Haney and Marc P. Christensen

The sliding-banyan (SB) network employs an interleaved multistage shuffle–exchange topology, implemented with a three-dimensional free-space interconnection architecture that connects a multichip backplane to itself. Surface-normal emitters and detectors, which compose the stages' input–output, are spatially multiplexed within the same chip location, along with electronic control and switching resources. A simple deflection self-routing scheme minimizes internal contention, providing efficient use of switching and interconnection resources. The blocking performance of the SB is quantified through simulations based on realistic nonuniform traffic patterns. Results show that the SB architecture requires significantly fewer resources than other self-routing banyan-based networks. The multistage-switching and interconnection-resource requirements are close to the theoretical minimum for nonblocking networks, and the SB's distributed self-routing control resources grow only approximately linearly with the number of nodes, providing good scalability. © 1997 Optical Society of America

*Key words:* Optical interconnects, switching networks, smart pixels.

## 1. Introduction

There is an ever increasing demand for high-throughput, cost-effective, broadband data-switching networks, as demonstrated by the explosive growth in the asynchronous transfer mode (ATM) equipment industry. Future networks must handle thousands of high-bandwidth channels, implying an aggregate capacity in the terabit per second regime.[1] Electronic switching approaches, based on high-speed VLSI and associated packaging, may not scale cost effectively above the 100-Gbits/s regime.[2] Alternative technologies and architectures will be needed to meet the coming demand.

In this paper the performance of the sliding banyan (SB) network[3,4] is analyzed. The SB is a multistage interconnection network (MIN) architecture based on a new three-dimensional (3-D) shuffle-interconnection topology in which multiple stages' input–output (I/O) resources are interleaved (spatially multiplexed) on a common backplane. With this topology, the critical I/O, switching, and control resources for a given node are placed in close proximity on the backplane, such that they are contained within the same optoelectronic integrated circuit (OEIC). With a suitable destination-tag self-routing control algorithm, the SB's physical arrangement permits rapid removal of correctly routed packets from the fabric.

Initial simulations and analysis have demonstrated the SB's advantages for permutation traffic[3,4] in which every input node is connected to exactly one output node, selected randomly. The SB was shown to provide a significant reduction in resources owing to its unique resource-partitioning scheme. This reduction stems from the SB's topology and efficient self-routing control strategy, in which each packet is effectively routed through a banyan that has "slid" in the time domain to accommodate that packet's needs. This results in a significant reduction in internal contention in the switch, with a commensurate reduction in switching and interconnection resources.

The focus of this paper is on the performance of the SB under a realistic and demanding traffic load. Real systems characteristically have nonuniform traffic distributions. For example, in modern data-communication systems, many nodes may require data from a single location (i.e., a server). In this situation many packets may have identical destinations. This is, perhaps, the worst-case scene, because the opportunity for internal contention is dramatically increased. Many topologies attempt to compensate for this type of internal contention through the use of redundant resources. The additional resources reduce the efficiency of the switch. However, as is shown in this paper, the SB is particularly impervious to this inefficiency, as a result of its ability to remove correctly routed packets from the switching fabric immediately. These benefits derive

The authors are with the Department of Electrical and Computer Engineering, George Mason University, Mail Stop 1G5, Fairfax, Virginia 22030-4444.

Received 5 February 1996; revised manuscript received 7 October 1996.

0003-6935/97/112334-09$10.00/0

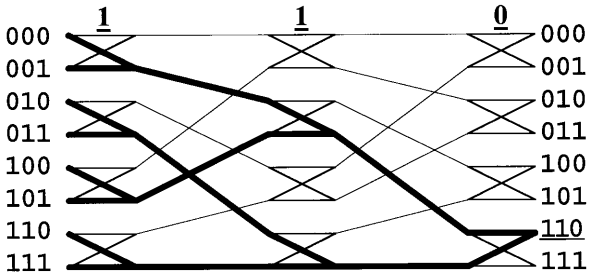


Fig. 1. Shuffle-interconnected banyan network for $N = 8$ nodes, consisting of $\log_2 N$ stages of $(N/2)$ $2 \times 2$ switches connected by $(\log_2 N) - 1$ identical PS interconnection patterns. The paths for destination-tag routing for all inputs to destination 110 are highlighted in boldface. The switch settings, corresponding to the bits of the destination address, appear above each sequential stage. A packet is switched to the lower node if the bit is a 1, and to the upper if it is a 0. This self-routing approach is independent of the packets' input node and its current location.

from the sliding time window made possible by the interleaved topology. This is a fundamental difference from other self-routing MIN's in which, because of limited I/O resources, packets are removed from the network only at the last stage, or possibly in a small subset of the stages.

In Section 2 we review the SB topology, routing-control algorithm, and optical interconnection scheme. In Section 3, we describe the simulation used to validate the SB's performance, along with the key results that include a comparison with other redundant banyan networks. In Section 4 we discuss the results and their impact on resource requirements and scalability for the SB. In the conclusion (Section 5), we discuss the implications of enhanced performance on a physical implementation.

## 2. Sliding-Banyan Network Architecture

### A. Banyan-Based Multistage Interconnection Networks

Any network in which there exists a unique path from any input to any output is called a banyan network.[5] One particular type of MIN-based banyan consists of a set of $\log_k N$ stages, each containing $N/k$, $k \times k$ crossbar switches interconnected in point-to-point butterfly or shuffle interconnection patterns. Figure 1 depicts a banyan network for $N = 8$ nodes, with $k = 2$. This type of banyan is particularly useful because simple destination-tag self-routing may be employed. As shown in the figure, each stage's $k \times k$ crossbar switch effectively sets $\log_2 k$ bits of the final destination address. When $k = 2$, each stage sets one bit. The banyan depicted in Fig. 1 is interconnected in a regular shuffle interconnection pattern—the pattern is identical between each stage of the banyan. Other interconnection patterns can achieve the same self-routing results. The type of interconnection describes the type of banyan. Some banyans have interconnections that differ between every stage (e.g., the butterfly), while others have the same interconnection pattern between all stages [e.g., the perfect shuffle (PS)[6]]. In any case, the routing of the packet is independent of the packet's input location. This simple self-routing approach minimizes the control resources for packet routing but suffers from possible internal packet contention—a fully loaded banyan cannot route all packets successfully. Packets need not be lost because of this, as there are always exactly $k$ inputs and $k$ outputs to each switch. A packet may simply be sent to an incorrect output.

Redundant banyans have been suggested to overcome internal contention.[7–9] One such architecture, the tandem banyan (TB),[8] is depicted in Fig. 2. In the TB unsuccessful packets are not blocked but rather tagged and misrouted to the end of the banyan. Packets that successfully arrive at their destinations are removed from the network, and the tagged packets' are reset and sent into another banyan. The reduced traffic on the next banyan makes it more likely that each packet will be successfully routed. Banyans are appended in this way until an arbitrarily low blocking probability is reached.

### B. Sliding-Banyan Concept

The SB architecture takes advantage of a different type of redundancy to handle internal contention more efficiently. Instead of a set of discrete banyans, the SB is composed of several banyans' worth of identical PS interconnections between stages, as depicted in Fig. 3. A deflection routing scheme is used

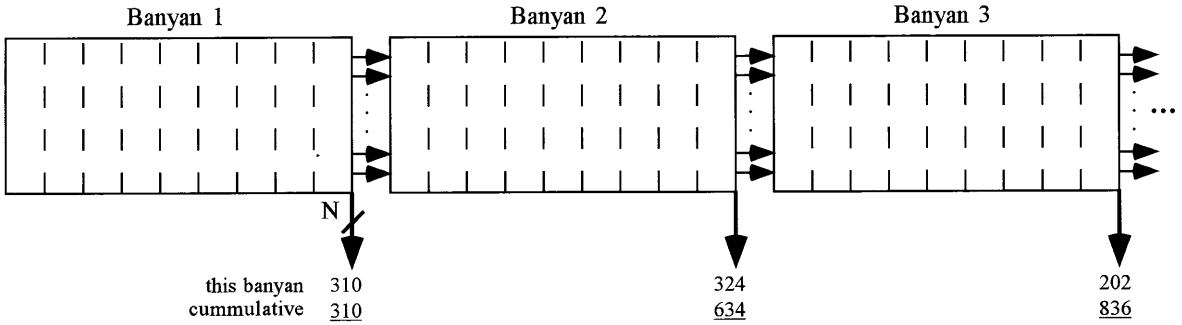


Fig. 2. TB concept: Banyans are 10 stages long, corresponding to $N = 1024$. Successfully routed packets exit the switch at the end of each banyan. Unsuccessfully routed packets are sent to the next banyan. The numbers indicate the current number and cumulative number of successfully routed packets at the end of each banyan for an actual simulation.

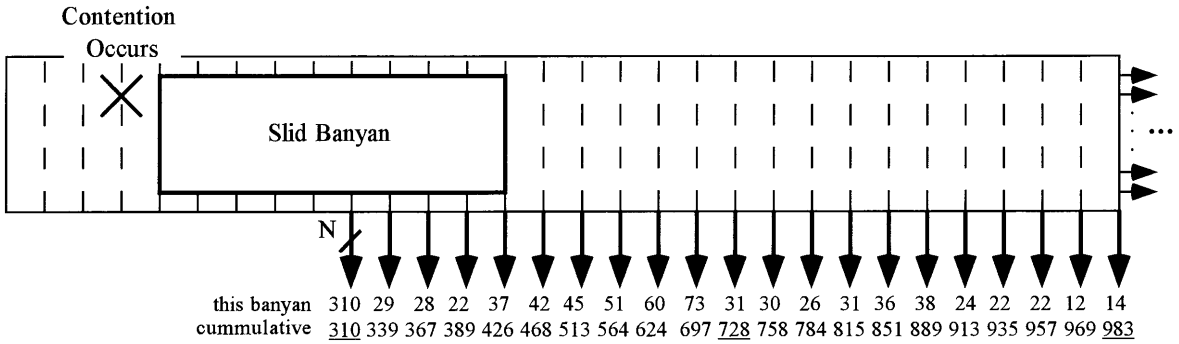


Fig. 3. SB concept: A collection of shuffle-interconnected stages, with $N$ outputs available at each stage after the first banyan's last stage. When an internal contention occurs, a virtual banyan slides to align with the next stage to accommodate the routing and removal of the lower-priority packet in $\log_2 N$ stages from that point. The numbers indicate the current number and cumulative number of successfully routed packets at the end of each stage for the same simulated traffic that generated the numbers of Fig. 2. The SB removes packets more rapidly than does the TB.

instead of the TB's bit-synchronous destination-tag routing. In the TB, the first bit is examined at the first stage of the banyan, the second at the second, and so on. In deflection routing, used in the SB, the examination of a bit is not tied to its location in a physical banyan. Instead, when a contention occurs, a virtual banyan *slides* to accommodate the immediate rerouting of the packet. Since all groups of $\log_k N$ stages are interconnected with the same pattern, this virtual banyan remains unaffected by the location of the contention. This routing scheme allows for rapid removal of the packets from the network but requires an output path at every stage of the network, instead of at the end of each banyan. This concept could become prohibitive for large networks if all the output paths were isolated output drivers. For example, if there were 1024 nodes ($N = 1024$) with 60 stages, there would be 60,000 output drivers needed for the implementation depicted in Fig. 3. This number is impractical for an all-electronic implementation because of the prohibitively large number of required electronic output drivers and metallic interconnections (e.g., coaxial cable).[10]

Since only a small fraction of the output drivers associated with any node can be utilized, it is advantageous to find a partition of the resources that removes this inherent redundancy in output drivers. This can be achieved by a partitioning scheme that spatially co-locates all stages of a given node, thereby allowing them all to have access to a much reduced number of output drivers. For a large switch, this is physically unrealizable in a VLSI-based all-electronic implementation as a result of the switching fabric's interconnection demands on resource partitioning. However, as shown in Subsection 2.C, using two-dimensional (2-D) arrays of surface-normal optical interconnections in an interleaved shuffle removes the undesirable output-driver redundancy.

### C. Optical Sliding-Banyan Topology

Three-dimensional shuffle-interconnection approaches provide a much higher bisection bandwidth capability than can be achieved by electronic packaging technologies. Several free-space optical shuffle-interconnection approaches have been proffered.[11–19] These approaches exploit the emerging technologies of low-power vertical-cavity surface-emitter laser (VCSEL) and detector arrays to distribute the I/O across the surface of the OEIC's instead of around the perimeter. The performance advantages of the SB stem from a new way of partitioning the resources, made possible by a retroreflective free-space optical shuffle-interconnection approach.

Figure 4(a) depicts a schematic side view of an optically interconnected MIN. The shuffle links are implemented with physically separated optical systems—one for each stage. This is the typical approach considered in optically interconnected MIN's. The optics shown implement a shuffle by means of a symmetric arrangement of lenses, requiring four lenses for each shuffle interconnect.[19] It has been pointed out, however, that the optical shuffle has a *local* shift-invariance property that makes it possible to interleave multiple stages' I/O's and implement all required shuffle links simultaneously with a single optical system.[20] Such an approach, as depicted in Fig. 4(b), provides the ability to pipeline effectively multiple stages within a single optical system. Furthermore, the symmetry of systems like the one depicted in Fig. 4(b) permit the shuffle optics to be implemented with a single array of lenses and a mirror.[3] This concept is illustrated in Fig. 4(c). The switching, control, and I/O resources of the multiple stages are repartitioned in an interleaved manner onto a single plane. All stages of a single node are co-located within that plane. If this co-location is provided on a single chip, then only one off-chip output driver is required for each node—a dramatic reduction from the requirements of Fig. 4(a). The new resource-partitioning scheme utilizes a single macro-optical shuffle-interconnection module, which is simultaneously used by all stages. The resources are therefore distributed laterally across a single backplane, rather than longitudinally across many planes. Such a planar distribution is amenable to

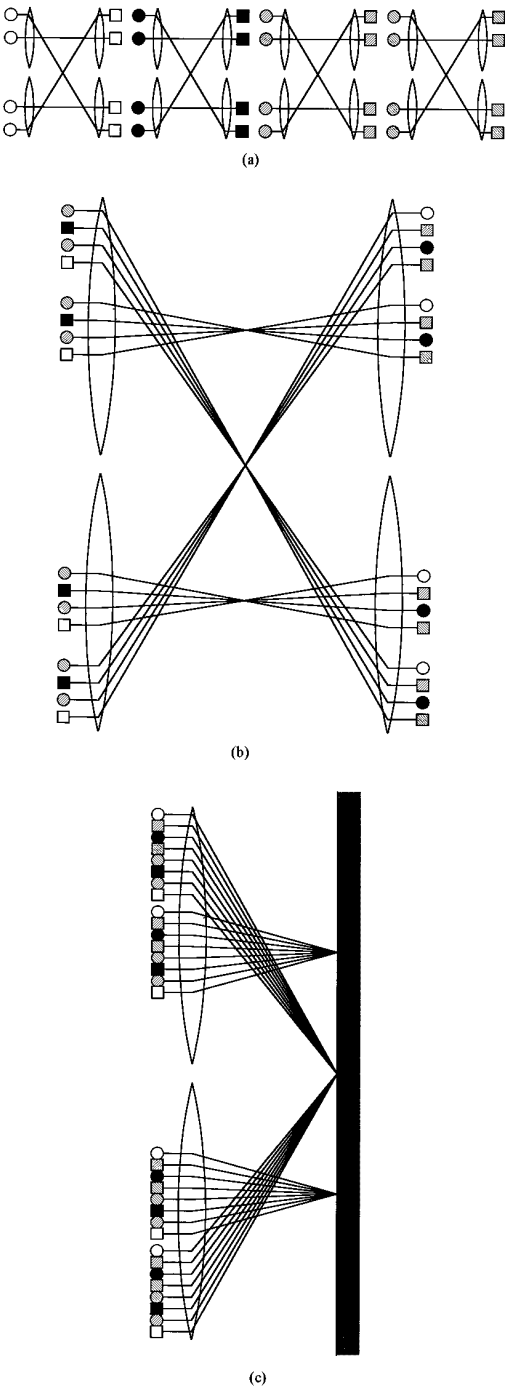


Fig. 4. (a) Schematic side view of an optically interconnected MIN. The circles represent emitters, and the squares represent detectors. (b) Schematic side view of the interleaved single optical system that performs the same interconnection pattern as (a). (c) Schematic side view of a folded version of (b), showing that all optoelectronics may be contained in clusters in a single plane.

implementation with conventional multichip module (MCM) packaging technology.

For the large (e.g., $N = 1024$ nodes) switching applications envisioned for the SB concept, the electronic switching and control functions are integrated with the optoelectronic I/O functions in an array of OEIC's distributed across an optoelectronic backplane containing many such chips. Each OEIC will handle a subset of the nodes. For example, a possible configuration might comprise 16 nodes/OEIC, with an $8 \times 8$ OEIC array on the backplane, to give a total of 1024 nodes. If there are 50 pipelined stages in the SB network, then each OEIC requires $50 \times 16 = 800$ optoelectronic I/O's. Several promising high-speed OEIC technologies, based on monolithic and hybrid integration of VCSEL's and detectors with logic, are now under development.[21] The SB's interleaved multichip optical shuffle approach combines the global interconnection advantages of optics with local low-power on-chip electronic interconnections to effect the appropriate links and avoid redundant output drivers.

### D. Sliding-Banyan Routing Control

High-aggregate bandwidth (e.g., 0.1–1 Tbytes/s) switches will require distributed local routing control for scalability. Self-routing shuffle-based networks utilizing a deflection algorithm to route packets to their destinations have been proposed.[22] The SB employs a variation of deflection routing that is based on simple destination-tag routing. In this scheme the headers of each set of $k$ (or fewer) packets entering a local $k \times k$ switch are examined to determine the proper switch settings. The simplest kind of contention resolution will occur for a value of $k = 2$. To simplify the explanation, let us assume that $k$ is 2 in the following discussion, although the basic idea can be extended to higher-order $k$-shuffle–exchange networks.

The basic banyan for the SB consists of $\log_2 N$ stages of PS interconnections, each of which is followed by a set of $N/2$, $2 \times 2$ crossbar switches. Figure 1 shows how a packet would be routed in one of the SB's banyans. The packet is first shuffled from its original address to the first set of switches. If there is no contention, then a switch controller examines the destination address's most significant bit (MSB) and routes the packet to the bottom output of the $2 \times 2$ switch if the bit is a 1 and to the top output if it is a 0. After the next shuffle stage, the next MSB is examined, and the switch is set appropriately, as in the first stage. At the final stage, the least significant bit (LSB) of the destination tag provides the last switch setting for that packet. If two packets require the same output node of the switch, only the packet with the highest priority is switched to that node, while the other is routed to the other node of the $2 \times 2$ switch and flagged. A flagged packet must restart its destination-tag routing to begin anew at the destination address's MSB. The effect of restarting is that of a new virtual banyan's being slid to begin at that packet's current location in the series of stages, as depicted in Fig. 3. If no further contention occurs, the rerouted packet will complete its routing through the slid banyan and then exit the network at its destination node.

Packets are assigned a priority tag according to their current degree of success at routing to deter-

mine which packets take precedence on internal contention. Packets that have nearly completed their routing have the highest priority. The degree of success is determined by how many *consecutive* stages the packet has successfully (i.e., without losing a switch-contention event) traversed. If a conflict occurs between two packets, the header decoder routes the packet with the higher priority correctly and flags the other packet to begin rerouting at the next stage.

When a packet reaches its destination, a simple on-chip switch is required to route it out of the switching fabric, since all stages of the pipelined network are physically co-located. Packets are thereby immediately routed to their destinations, without the burden of excess underutilized switching resources, complicated control, or a large number of power-hungry high-speed output drivers. The distributed nature of the SB's routing control algorithm provides the means to high aggregate throughput, while the simple logic functions needed for the SB indicate that the overall overhead for control functions in the SB will be low. A detailed trade-off analysis for various routing control algorithms is introduced in Ref. 23 and is the subject of continuing analysis.

The number of stages needed in the SB is determined by the overall blocking performance desired. Simulations and a blocking-rate model were developed to evaluate the SB's performance under full permutation-traffic conditions.[4] The model's predictions for the blocking-error rate followed very closely the actual data achieved by compilation of statistics over many simulation runs. Both PS ($k = 2$) and four-shuffle configurations were evaluated for the TB and SB. For $N = 1024$ and $k = 4$, it was found that the SB required 30 stages to achieve an overall blocking probability of $10^{-12}$, whereas the TB required 45 stages—a significant improvement in switching and interconnection resources and latency.

The key advantage demonstrated by the SB is the *immediate* rerouting of deflected packets. In the TB, when contention occurs the deflected packet must be pushed along to the end of a banyan before it can begin its rerouting process. This is necessary because of the limited number of exit points the packet has available to it in the TB architecture. In the TB a packet can complete its routing only at the end of a banyan, therefore it can begin routing only at the beginning of a banyan. This results in many switching stages being utilized for passing a deflected packet along to the end of the banyan, instead of routing the packet to its destination. The numbers next to the exit lines of Figs. 2 and 3 show how immediate rerouting of packets affects the load on the switching fabric for a typical simulation run. In the TB, no packets exit the network within banyans, whereas in the SB packets continually exit the network after the first banyan.

In the TB groups of $\log_k N$ stages (banyans) are appended to the end of the switching fabric until the desired arbitrarily low blocking probability is reached. This large commodity of stages can be wasteful, as the last banyan in a network seldom has more than one packet on it. The SB adds *single* stages as necessary for acquiring an arbitrarily low blocking rate. In this way, the SB maintains a resource advantage over the TB for all blocking probabilities.

### 3. Area-of-Interest Traffic Analysis

The analysis of permutation traffic provided a good first indication of the advantages of the SB. However, more realistic nonuniform traffic patterns, such as area-of-interest (AOI) traffic, are more challenging to switching performance and must be evaluated. AOI traffic patterns result when many packets are routed to the same destination. This destination may be several contiguous nodes or simply a single node. For AOI traffic, it is assumed that output buffering is used such that all packets that are not lost *within* the switch will be successfully passed on to the output, so output contention is avoided. When a packet reaches its destination switch it is allowed to exit, even if the final switch has contention. An output buffer is therefore required to handle up to $L$ packets simultaneously, where $L$ is the number of network stages with output ports. Such a buffer, which would be required for any switch that can handle AOI traffic, will require a nontrivial design in practice. The buffer design will likely require an upper limit on the size of the AOI. For the SB, it is anticipated that some on-chip multiplexing of shared buffer resources will be used to reduce the required buffer redundancy.

Even with the assumption of no output contention, internal contention may diminish the overall performance of the switch for AOI traffic. Since many packets may have the same or similar destination tags in their headers, there will be an increased probability that packets will experience collisions internally. For any given fixed switching architecture, AOI traffic will likely increase the overall probability of blocking for the switch. The selection of an architecture should be driven by the efficiency and robustness to the variations in traffic that may be encountered. A robust switching architecture will have very little change in efficiency or resource utilization under variations in traffic patterns.

To determine robustness and efficiency, the average number of stages necessary to route all packets successfully was calculated for various AOI traffic patterns. The performance of the SB was compared with that of various types of TB's. From these data, the total amount of switching and interconnection resources was estimated and the overall resource utilization deduced. The program simulated the routing algorithms of the various switches, measured how many stages are needed to route successfully all of the inputs, and then compiled performance statistics for many thousands of runs. In particular, the SB was compared with several replicated versions of the TB under AOI traffic between 0% and 3%. In this case a TB with a replication factor of $q$ was simulated by $q$ TB's, with a $1/q$ traffic load for each.

The simulated AOI traffic was generated as follows:

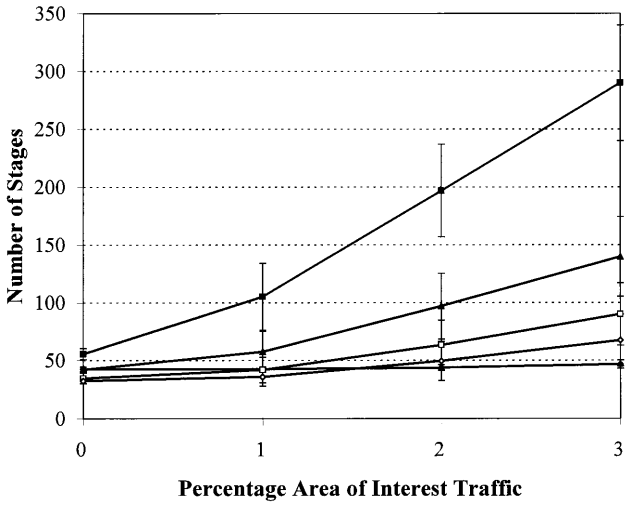


Fig. 5. Comparison of the SB with the TB and replicated versions of the TB (TB-RN) for AOI traffic. The replicated TB's have 2, 3, and 4 TB's in parallel, with one half, one third, and one fourth of the traffic load, respectively, in each simulated run. The vertical axis represents the number of stages required to route all 1024 packets. The solid squares represent the TB alone; the solid triangles, TB-R2; the open squares, TB-R3; the open diamonds, TB-R4; and the solid diamonds, the SB.

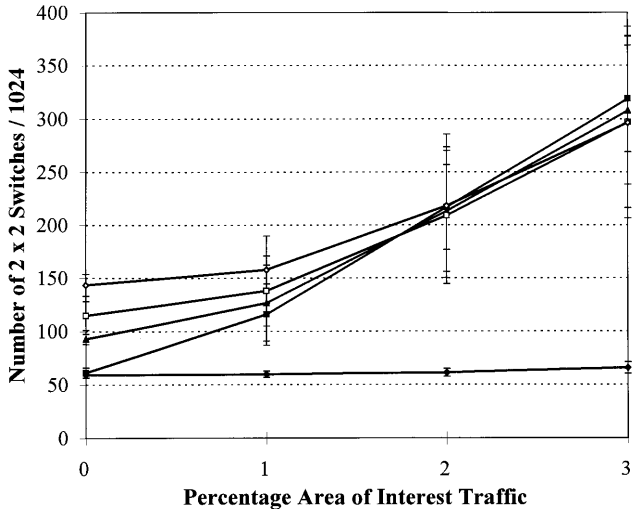


Fig. 6. Data of Fig. 5 normalized to take into consideration the redundant resources of replicated stages and output switching. The vertical coordinate represents the switching resources required to route all 1024 packets. The open diamonds represent the TB-R4; the open squares, the TB-R3; the solid triangles, the TB-R2; the solid squares, the TB alone; and the solid diamonds, the SB.

First, a random permutation of the input set was generated. Then a fixed percentage (between 0% and 3%) of those addresses was set equal to a single, randomly picked, output node. The area consists of a single node in this case. The same traffic pattern was presented to the SB and various replications of the TB. Figure 5 represents the results of 1000 simulated runs of the SB and TB under AOI traffic. The mean number of stages required is plotted, with error bars indicating the standard deviation of the data. The TB data have an apparently larger error caused by the fact that the possible number of stages were integral groups of $\log_2 N$ stages.

As shown in Fig. 5, the AOI traffic has very little effect (less than 3–5 additional stages, or approximately 10%) on the SB. However, AOI traffic significantly increases the number of stages of a single TB, requiring well over 200 more stages for 3% AOI. The replicated TB's suffers less from this effect. The replicated TB's achieve this by a decrease in the length (latency) of the switching architecture at the expense of its breadth. Although this replication has a beneficial effect on the number of stages required for this traffic pattern in the TB, each stage now has several times the number of switching resources as a single stage of the TB or SB. These added resources are accounted for by normalization of the data in Fig. 5 by the total number of switching resources. The replicated TB's have, in effect, traded resources for decreased latency. Yet, in ATM switching, relative latency becomes an issue only when it approaches the time length of a packet; if the latency is less than a packet length, misordering of the data is avoided.

Figure 6 shows the results of such normalization of the data. These data show that, although replications of TB's improve latency, the overall switching resources remain unaffected. The switching-resource requirements grow at an undesirable rate under increasing AOI traffic, independent of the degree of replication. In this analysis, switching requirements also include output switching resources, i.e., those necessary for removal of packets. In the SB this occurs at every stage of the network, whereas in the TB it occurs only at the end of each banyan. Although the SB pays a heavier penalty in switching to exit the network, the penalty is insignificant compared with the penalty paid by the redundant resources of the TB. In fact, the redundant versions of the TB perform equally poorly under this traffic pattern when resource utilization is examined. The resource requirements for the SB remained nearly constant over all traffic conditions studied. It was verified in further simulations that the trends (slopes) depicted in Figs. 5 and 6 continue for all higher percentages of AOI.

## 4. Discussion

The simulation results clearly show that the SB is relatively immune to AOI traffic effects, whereas the TB's have great difficulty with AOI traffic under any replication scheme. This difficulty is due to the TB's bitwise synchronous routing scheme: The first bit is examined at the first stage, The second bit at the second stage, and so on. If many packets are destined for the same area, most of their bits will be identical; if they are destined for the same node, as in the above simulations, the addresses are identical. After deflection occurs, the remaining packets all collide again and again on the same stages of each successive banyan, letting only a few pass each time. If

we recall that 1% of 1024 node network is approximately 10 packets and only two at most can succeed at each banyan, then it is not surprising that this traffic pattern is difficult for the TB.

Replicated TB's reduce the latency of packet routing, allowing several times more AOI packets to be routed in each banyan. This somewhat reduces the problem of bitwise synchronous blocking in AOI traffic. This recurring bitwise synchronous blocking is similar to a problem faced in the ALOHA communications protocol[24] in that two packets may continually interfere with each other and prevent successful routing. The accepted solution to this problem for the ALOHA is to introduce a random delay when potential blocking occurs. This delay desynchronizes the packets and allows them to continue without continually colliding. This solution is analogous to how the SB avoids bitwise synchronous blocking. When a packet is deflected in the SB, it begins rerouting immediately. This immediate rerouting aligns the bitwise decoding of the packet's header with its current stage (where the error occurred). This action spreads the packets headed for the AOI among several nonsynchronous (i.e., noninterfering) banyans.

Similar simulation comparisons resulted between the SB and TB for different numbers of nodes. For example, it was found that whether $N = 512$, $N = 1024$, or $N = 2048$, the same relative efficiencies were obtained. This result therefore shows that the SB's resource-efficiency performance scales well with network size relative to the various TB variations tested.

Since the simulations provided a count of the actual switching and interconnection resources necessary to achieve good blocking rates, the absolute resource efficiency of the SB can also be characterized. For example, it is well known that the Benes network uses $(\log_2 N) - 1$ stages, which is the minimum number of MIN switching and interconnection resources needed to achieve a rearrangeably nonblocking network.[5] However, the Benes has been found to be entirely impractical for large (many nodes), high-throughput applications as a result of limitations on the necessary control algorithm. The implementation of a nonblocking network in the Benes concept requires a *global* control algorithm. No efficient (nonrecursive or linear) algorithm is known for setting the switches. Furthermore, the Benes has no provisions for handling AOI traffic. The SB simulation results above show that typically less than $5 \log_2 N$ stages are required to route all packets successfully with high AOI traffic. With the additional overhead of output switching, the total switching resources remain within a factor of 4 of the Benes network. Yet the SB maintains the critical practical advantages of distributed self-routing control and the ability to handle nonuniform traffic.

For investigating the algorithmic efficiency of a MIN architecture, the switching-resource efficiency may be examined. In the SB and TB, switching-resource usage falls into three categories. A switching resource was either used to route a packet correctly, used to pass a packet to the next stage to later be routed, or not used at all. Figure 7 plots the average percentage (over 1000 random simulation runs) of switching resources utilized for correctly routing packets in each of the networks. The SB remains efficient over the range of AOI traffic patterns, whereas the TB and its replications become inefficient. The ideal would be 100% switching utilization efficiency. The Benes network, the minimum size rearrangeable nonblocking network, would be 100% efficient for permutation traffic under this definition, but, as mentioned above, is impractical because of control issues and the fact that it cannot effectively handle AOI traffic.

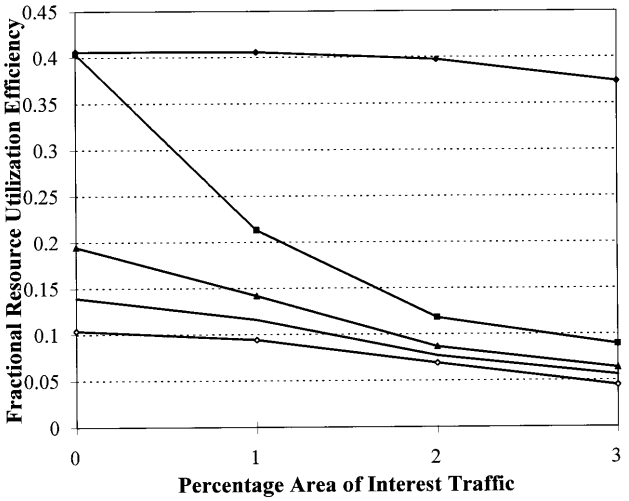


Fig. 7. Comparison of switching-resource utilization efficiency for the SB, the TB, and several replicated versions of the TB. Only switches that contribute to the correct routing of a packet counted for this measure. The SB maintains high switch-resource utilization efficiency for a wide spectrum of nonuniform traffic. The solid diamonds represent the SB; the solid squares, the TB alone; the solid triangles, the TB-R2; curve with no symbols, the TB-R3; and open diamonds, the TB-R4.

In the analysis in Section 3, we focused on the switching- and interconnection-resource requirements of the SB concept. However, a complete analysis must address all types of resources necessary to implement the SB. The resources necessary to implement any switching concept are broken down into three general categories: switching, interconnection, and control. The switching resources are further broken down into internal switching (routing of packets) and output switching (removal of packets). The interconnection resources are also further subdivided into drivers and links. Here, drivers includes electronic or optoelectronic interchip drivers. Interchip drivers are assumed to be a dominant source of resource requirements because intrachip interconnections are typically much lower in power consumption. The link resources comprise physical interchip, interboard metallic paths, or both, such as high-speed lines and coaxial cable, needed to implement the desired shuffle–banyan link patterns. For free-space interconnection systems, such as the one employed in the

Table 1. Resource Requirements for the SB with $\sim 10^{-6}$ Probability of Blocking

| Resource | Requirement |
|---|---|
| Switching: internal | $5(2 \times 2)(N/2)\log_2 N$ |
| Switching: output | $4N \log_2 N$ |
| Interconnects: chip output drivers | $N$ |
| Interconnects: free-space links | $N(5 \log_2 N - 1)$ |
| Control | $\propto N$ |

SB, the link resources are composed of the optical elements and free-space volume necessary to achieve the required shuffle links.

For good scalability a switch concept must exhibit reasonable resource growth requirements, with throughput for *all* of the main types of resources delineated above. Table 1 summarizes the growth requirements for the SB. In the SB, the internal switching resources are the actual local electronic switching elements necessary to effect the SB routing algorithm. The switching requirements for the internal switching is $5(2 \times 2)(N/2)\log_2 N$. This corresponds to five banyan's worth ($5 \log_2 N$) of stages, each with $N/2$ switches of complexity $2 \times 2$. The coefficient 5 comes directly from the blocking probability; a different blocking probability would simply lead to a different coefficient. Previous studies have shown that a small increase in this coefficient results in a dramatic reduction of the probability of blocking. For example, for a change from a $10^{-6}$ to a $10^{-30}$ blocking probability, the coefficient increases from 5 to 7. The minimum number of switching resources is $(2 \times 2)(N/2)[(2 \log_2 N) - 1]$ for the Benes network—but this requires complex control.

The output switching resources in the SB stem directly from the ability to remove packets at any stage after the first banyan. Therefore, the output switching requirement is $4N \log_2 N$. This corresponds to adding $N$ output paths to forth banyan's worth of stages to allow packets to be switched out of the switching fabric. The number 4 is tied directly to the length of the network; it will always be one less than the coefficient for internal switching, since the first banyan requires no internal output paths.

The unique partitioning of the SB reduces the number of output drivers to $N$, one for each node of the network, whereas an approach with physically separated stages (as would be required in a VLSI implementation) would require $5N \log_2 N$. The optical shuffle link requirement is equal to the product of the number of SB stages (minus 1) and $N$ nodes, since each pair of stages has $N$ interconnects between them.

Practical limitations on the interleaved-shuffle interconnection approach are determined by the quality of the optical elements and the physical optoelectronic I/O spacing requirements on and between the smart-pixel OEIC's.[25–27] Preliminary experiments with VCSEL arrays and wide-angle imaging shuffle lenses have validated the basic retroreflective approach,[4] and a full prototype was demonstrated with a high-resolution mask that simulated a high-density multichip VCSEL–detector array.[28]

Finally, the distributed control of the SB allows a nearly linear growth in control resources as only one processor per node is required to implement routing for all stages for that node. The use of a single control processor for each node is achieved by means of the unique 3-D topology of the SB architecture; without the co-location of stages, this linear growth would not be feasible.

## 5. Conclusion

The realization of a SB switch with 0.1–1-Tbyte/s aggregate throughput[26] will hinge on research in three related areas. The first area concerns the selection of smart-pixel technology, distributed across many OEIC's on the SB's backplane, in which the required active resources will reside. Several promising monolithic and hybrid technologies in integrating electronic logic with arrays of emitters and detectors are rapidly emerging.[21] All these approaches are pushing toward the VLSI density of logic required for the SB's smart pixels. Previous estimates of the required optoelectronic I/O density on the smart-pixel OEIC's[5] of several hundred per centimeter appear to be within reach of the smart-pixel developers.

The second area concerns the design and optimization of the control algorithm and its implementation with a smart-pixel technology. Several variations on the basic SB control strategy are currently being evaluated. These variations include an analysis of the degree ($k$) of the local switch. The analysis thus far has usually assumed $k = 2$, providing the simplest SB-node functional logic. However, higher-order shuffle approaches will reduce the number of stages by approximately a factor of $k/2$, thereby reducing significantly the number of stages and optoelectronic I/O at the expense of more complicated nodes. These trade-offs must be evaluated to determine the optimum SB smart-pixel configuration.

The last area concerns the experimental validation of the retroreflective interleaved optical shuffle interconnection approach. An experimental SB optical module, similar to the one depicted schematically in Fig. 1, is currently under development.[26,28] It consists of a $4 \times 4$ array of identical high-quality miniature projection lenses that have a wide and flat field of view, with high resolution. Preliminary testing of the experimental module shows that it will be capable of implementing a full banyan interconnection pattern for a 256-node array.

The AOI traffic analyses described in this paper demonstrated the benefits of the SB topology. Significant improvement over other types of redundant-banyan approaches was demonstrated in total switching and interconnection resources required, as well as in latency. The performance enhancements of the SB stem from its unique partitioning of resources in a new 3-D optically interconnected topology in which a simple self-routing algorithm effects

the rapid routing and removal of packets from the fabric. The SB's robustness to nonuniform traffic and its linear or nearly linear resource growth with the number of nodes suggest that this concept will scale well with network size.

This research is sponsored by the Advanced Research Projects Agency through a contract with the Air Force Office of Scientific Research.

### References


1. J. Hui, "Switching integrated broadband services by sort-banyan networks," Proc. IEEE **79,** 145–154 (1991).
2. T. Egawa, K. Yukimatsu, and K. Yamasaki, "Recent research trends and issues in photonic switching technologies," NTT Rev. **5,** 30–37 (1993).
3. M. W. Haney and M. P. Christensen, "Optical freespace sliding tandem banyan architecture for self-routing switching networks," in *Technical Digest of the International Conference on Optical Computing*, 22–25 August 1994, Edinburgh, Scotland, pp. 249–250.
4. M. W. Haney and M. P. Christensen, "Sliding banyan network," J. Lightwave Technol. **14,** 703–710 (1996).
5. F. Thomson Leighton, *Introduction to Parallel Algorithms and Architectures: Arrays, Trees, Hypercubes* (Morgan Kaufmann, San Mateo, Calif., 1992).
6. H. S. Stone, "Parallel processing with the perfect shuffle," IEEE Trans. Comput. **C-20,** 81–89 (1971).
7. C. P. Kruskal and M. Snir, "The performance of multistage interconnection networks for multiprocessors," IEEE Trans. Comput. **C-32,** 1091–1098 (1983).
8. F. A. Tobagi, T. Kwok, and F. M. Chiussi, "Architecture, performance, and implementation of the tandem banyan fast packet switch," IEEE J. Select. Areas Commun. **9,** 1173–1193 (1991).
9. A. V. Krisnamoorthy and F. E. Kiamilev, "Fanout, replication, and buffer sizing for a class of self-routing packet-switched multistage photonic switch fabrics," in *Photonics in Switching*, Vol. 12 of OSA 1995 Technical Digest Series (Optical Society of America, Washington, D.C., 1995), paper PThC4-1, pp. 87–89.
10. T. J. Cloonan, "Comparative study of optical and electronic interconnection technologies for large asynchronous transfer mode packet switching applications," Opt. Eng. **33,** 1512–1523 (1994).
11. A. Lohmann, W. Stork, and G. Stuke, "Optical implementation of the perfect shuffle," in *Topical Meeting on Optical Computing*, Vol. 5 of OSA Technical Digest Series (Optical Society of America, Washington, D.C., 1985), paper WA3.
12. A. W. Lohmann, "What classical optics can do for the digital optical computer," Appl. Opt. **25,** 1543–1549 (1986).
13. G. Eichmann and Y. Li, "Compact optical generalized perfect shuffle," Appl. Opt. **26,** 1167–1169 (1987).
14. S.-H. Lin, T. F. Krile, and J. F. Walkup, "2-D optical multistage interconnection networks," in *Digital Optical Computing*, R. Arrathoon, ed., Proc. SPIE **752,** 209–216 (1987).
15. K.-H. Brenner and A. Huang, "Optical implementations of the perfect shuffle interconnection," Appl. Opt. **27,** 135–137 (1988).
16. C. W. Stirk, R. A. Athale, and M. W. Haney, "Folded perfect shuffle optical processor," Appl. Opt. **27,** 202–203 (1988).
17. A. A. Sawchuk and I. Glaser, "Geometries for optical implementations of the perfect shuffle," in *Optical Computing '88*, P. Chaval, J. W. Goodman, and G. Roblin, eds., Proc. SPIE **963,** 270 (1988).
18. M. W. Haney and J. J. Levy, "Optically efficient free-space folded perfect shuffle network," Appl. Opt. **30,** 2833–2840 (1991).
19. G. C. Marsden, P. J. Marchand, P. Harvey, and S. C. Esener, "Optical transpose interconnection system architectures," Opt. Lett. **18,** 1083–1085 (1993).
20. M. W. Haney, "Pipelined optoelectronic free-space permutation network," Opt. Lett. **17,** 283–285 (1992).
21. *Proceedings of the IEEE/LEOS Topical Meeting on Smart Pixels* (IEEE-Lasers and Electro-Optics Society, Boston, Mass., 1994).
22. M. Decina, F. Masetti, A. Pattavina, and C. Sironi, "Shuffleout architecture for ATM switching," in *Proceedings of ISS '92* (Institute of Electronics, Information, and Communication Engineers, Tokyo, Japan, 1992), paper A7.5.
23. M. W. Haney, C. B. Osborne, and M. P. Christensen, "Smart pixel algorithmic tradeoffs for the sliding banyan network," *Digest of the IEEE-LEOS Summer Topical Meeting on Smart Pixels* (IEEE-Lasers and Electro-Optics Society, Boston, Mass., 1996), pp. 107–108.
24. M. Schwartz, *Telecommunications Networks: Protocols, Modeling, and Analysis* (Addison-Wesley, Reading, Mass., 1988).
25. M. W. Haney, "Self-similar grid patterns in free-space shuffle/exchange networks," Opt. Lett. **18,** 2047–2049 (1993).
26. M. W. Haney and M. P. Christensen, "Optoelectronic sliding banyan network," U.S. patent 5,467,211, 14 November 1995.
27. M. W. Haney and M. P. Christensen, "Performance analysis and optical interconnection module evaluation for the free space sliding banyan network," in *Photonics Switching '96*, 21–25 April 1996, Sendai, Japan, pp. 90–91.
28. R. R. Michael, M. P. Christensen, and M. W. Haney, "Experimental evaluation of the 3-D optical shuffle interconnection module of the sliding banyan architecture," J. Lightwave Technol. **14,** 1970–1978 (1996).