A Holistic Approach to Designing Energy-Efficient Cluster Interconnects

Eun Jung Kim, *Member*, *IEEE*, Greg M. Link, *Student Member*, *IEEE*, Ki Hwan Yum, *Member*, *IEEE*,
N. Vijaykrishnan, *Member*, *IEEE*, Mahmut Kandemir, *Member*, *IEEE*,
Mary J. Irwin, *Fellow*, *IEEE*, and Chita R. Das, *Fellow*, *IEEE*

Abstract—Designing energy-efficient clusters has recently become an important concern to make these systems economically attractive for many applications. Since the cluster interconnect is a major part of the system, the focus of this paper is to characterize and optimize the energy consumption in the entire interconnect. Using a cycle-accurate simulator of an InfiniBand Architecture (IBA) compliant interconnect fabric and actual designs of its components, we investigate the energy behavior on regular and irregular interconnects. The energy profile of the three major components (switches, network interface cards (NICs), and links) reveals that the links and switch buffers consume the major portion of the power budget. Hence, we focus on energy optimization of these two components. To minimize power in the links, first we investigate the dynamic voltage scaling (DVS) algorithm and then propose a novel dynamic link shutdown (DLS) technique. The DLS technique makes use of an appropriate adaptive routing algorithm to shut down the links intelligently. We also present an optimized buffer design for reducing leakage energy in 70nm technology. Our analysis on different networks reveals that, while DVS is an effective energy conservation technique, it incurs significant performance penalty at low to medium workload. Moreover, energy saving with DVS reduces as the buffer leakage current becomes significant with 70nm design. On the other hand, the proposed DLS technique can provide optimized performance-energy behavior (up to 40 percent energy savings with less than 5 percent performance degradation in the best case) for the cluster interconnects.

Index Terms—Buffer design, cluster interconnect, dynamic voltage scaling, dynamic link shutdown, energy optimization, link design, switch design.

1 Introduction

Widespread use of cluster systems in a diverse set of applications has spurred significant interest in designing such servers, with performance, scalability, and quality-of-service (QoS) as the primary design objectives. In addition to these objectives, energy consumption in these architectures has recently emerged as a major concern since server power usage is becoming a significant fraction of the total ownership cost [1]. The energy consumption is critical as it affects the cost of cooling and backup power generation. In fact, new data centers in the Seattle area are forecast to increase the city's power demands by 25 percent [2]. Similar concerns have been raised for cluster-based data centers in other locations such as New York and California [1]. Recent technology trends in terms of power density limitations and design of compact and cheap cooling systems also motivate the need for energy efficient clusters in a single box or single board. Moreover, most designers are provided with a tight power budget that is becoming increasingly difficult to meet with the demand for more compact and higher performance clusters. All these factors have opened a new avenue of research in designing energy efficient, high-performance clusters.

While a large body of literature is available on power-aware processor and memory designs, network power optimization is an almost unexplored area of research. Cluster interconnects consisting of switches, links, and network interface cards (NICs) are used in many applications, connecting multiple PCs, multiple blades in a single server (Mellanox Nitro II [3]), multiple processors on a single board (Compaq Alpha 21364 [4]), or even multiple components on a single chip.

For example, Google operates its search engine on a cluster of more than 15,000 PCs connected by an interconnection fabric. In contrast, the Mellanox Nitro II Server Blade System houses up to 12 blade servers in a single box and uses the InfiniBand interconnection network [3]. Multiple Compaq Alpha 21364 microprocessors (each processor contains a processor core and integrated switch) can be used to form a network configuration on a single board to serve a variety of applications such as Web servers and telecommunication servers [4]. As the concept of networks-on-a-chip gains acceptance, it is anticipated that such interconnection fabrics will become more prevalent, even on a single chip.

It has been observed that the interconnection fabric consumes a significant portion of the total cluster power. For example, the integrated switch in the Alpha 21364 is

- *E.J. Kim is with the Department of Computer Science and Engineering, Dwight Look College of Engineering, Texas A&M University, H.R. Bright Building, Room 427C, College Station, TX 77843-3112. E-mail: ejkim@cs.tamu.edu.*
- *G.M. Link, N. Vijaykrishnan, M. Kandemir, M.J. Irwin, and C.R. Das are with the Department of Computer Science, Pennsylvania State University, 111 IST Building, University Park, State College, PA 16802. E-mail: {link, vijay, kandemir, mji, das}@cse.psu.edu.*
- *K.H. Yum is with the Department of Computer Science, The University of Texas at San Antonio, SB 3.02.05C, 6900 North Loop 1604 West, San Antonio, TX 78249. E-mail: yum@utsa.edu.*

Manuscript received 13 Feb. 2004; revised 16 Aug. 2004; accepted 8 Oct. 2004; published online 15 Apr. 2005.
For information on obtaining reprints of this article, please send e-mail to: tc@computer.org, and reference IEEECS Log Number TCSI-0049-0204.

0018-9340/05/$20.00 © 2005 IEEE Published by the IEEE Computer Society

reported to consume 20 percent of the chip budget, while 33 percent of the router linecard power is consumed in the interconnect in the Avici switch [5]. Similarly, the routers and links in a Mellanox server blade consume almost the same power as that of a processor (15W), roughly 37 percent of the total power budget [3]. These numbers indicate that power dissipation in the interconnect is significant and requires careful investigation. The interconnect fabric is composed of routers and links. The routers buffer incoming packets, determine their next destination, and then buffer them in the appropriate outgoing port. The links work at the physical layer of the network model and are comprised of a transmitter, a channel, and a receiver. While routers often consume much more chip area than links, as links traditionally include off-chip elements, the capacitance and power consumption of links is often significant.

A handful of prior studies have focused on modeling, characterizing, and optimizing the network energy consumption. The power consumption behavior and models of different switch fabrics have been explored in [6]. Techniques for optimizing power dissipation in high-speed links have been proposed in [7], [8]. Analytical power models for interconnection networks have been developed based on transistor counts [9], [10]. Wang et al. have presented an analytical power model to explore different switch configurations [11]. Different on-chip and chip-to-chip interconnect configurations are compared in [12] using an analytical approach. Shang et al. have extended the dynamic voltage scaling (DVS) technique to optimize link power in regular interconnection networks [13]. It was shown that DVS can conserve a significant amount of link energy at the expense of network performance. Concurrently, Soteriou and Peh have discussed a method of energy savings they term "On/Off Links" where individual links are shut down to conserve energy [14]. Their work is fundamentally different from ours in that their algorithm is designed to work on 2D mesh topologies, while we present an algorithm that functions on even irregular network topologies.

However, none of the prior efforts has taken a holistic approach to analyzing the power dissipation issues in various components of an interconnect. For example, the power simulator model in [12] considers energy analysis of the internals of a switch and links without any design optimization. On the other hand, the DVS model proposed in [13] optimizes only the links. While these models are elegant in their own right, they provide little insight about the overall energy issue. We believe it is essential to focus on a comprehensive power dissipation analysis of the entire interconnect when exploring energy optimization techniques in different components.

The research presented in this paper is an attempt in this direction. First, we perform a comprehensive power estimation of the complete communication substrate to understand the relative energy consumption in the switches, links, and NICs using the current 180nm technology. Second, since the links and switch buffers consume a significant amount of energy compared to other components, we propose circuit-level link and buffer organizations that can be used to determine accurate energy consumption. We explore two design optimizations to conserve energy consumption in the links; DVS and dynamic link shutdown (DLS). Our DVS policy, similar to [13], relies on the link utilization history and reduces the link frequency in seven steps. It captures clock synchronization overhead and buffer energy consumption in detail. The proposed DLS scheme, on the other hand, is based on the premise that if we can identify a subset of highly used links that can provide connectivity in the network, we should be able to completely shutdown other links if their utilizations are below a certain threshold. This leads us to our fourth contribution, where we exploit routing adaptivity to intelligently use a subset of links for communication, thereby facilitating dynamic link shutdowns and minimizing energy consumption. We use the InfiniBand Architecture (IBA) [15] style cluster interconnects that support both regular or irregular networks. For irregular networks, we use the shortest path first (SPF) algorithm that is known to provide better performance [16] and is compatible with the Internet routing scheme. For regular 2D networks, we use the well-known X-Y routing [17]. Finally, in order to quantify the impact of the switching protocols, we compare the overall performance and power dissipation in packet switching and wormhole switching paradigms.

We have designed a simulation tool that provides a cycle-accurate performance model for InfiniBand architecture (IBA) [15] style system area networks (SANs) and have incorporated energy parameters from actual designs for power characterization. We have simulated 15-node irregular networks and (8×8) regular networks with different workload parameters to characterize the energy-performance behavior with various energy optimization techniques. We measure average network latency and power consumption in different components. Our analysis reveals that the links and the buffers are the energy hot spots in the network and, thus, need careful design for reducing energy consumption. In this context, DVS can provide significant power saving at the expense of a high-performance penalty in low to medium workload situations. The average network latency degradation varies from 500 percent to 10 percent as the network load changed from 20 percent to 60 percent. Thus, the advantage of DVS gradually diminishes with network load. Further, we observe that, as technology scales to 70nm, the energy saving in DVS suffers due to the large amount of energy lost due to current leakage in the buffers. On the other hand, the proposed DLS technique can provide moderate energy savings with minimal degradation in average network latency. When combined with a suitable adaptive routing algorithm, the proposed technique can optimize performance and power over the entire workload. Integration of both DVS and DLS results in the best energy optimization. Finally, comparison between packet switching and wormhole switching techniques reveals that, while wormhole switching is a better option for high performance, it may not be as energy-efficient as the packet switching technique at higher loads.

The organization of the paper is as follows: In Section 2, the cluster interconnect architecture is discussed. The proposed energy optimization techniques are presented in Section 3. In Sections 4 and 5, the experimental platform and the simulation results are discussed, followed by the concluding remarks in Section 6.

2 System Architecture

The cluster interconnect used in this work is based on the IBA specification. In this section, we describe the switch fabric,

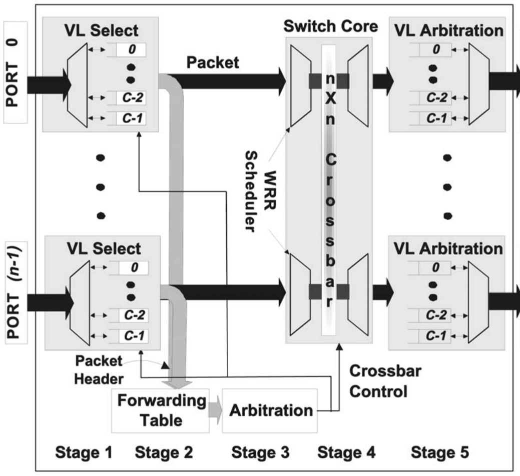


Fig. 1. A pipelined switch architecture.

NIC, and link architectures that are the main components of the interconnect and are modeled in this study.

2.1 Switch Architecture

The n-port switch modeled in this study adopts a five-stage pipelined packet-switched model, as shown in Fig. 1. The model can be changed to capture wormhole switching or virtual cut-through switching easily. The pipelined model represents the recent trend in router design [5].

Our IBA compliant switch design supports virtual lanes (VLs) [18] that provide a mechanism to implement multiple logical flows over a single physical link. The IBA specification allows between 2 and 16 VLs.

In the first stage, the incoming packets are assigned to one of the C VLs using the service level (SL) information in the packet header. The different SLs are mapped onto appropriate VLs based on a programmable lookup table. The header of a packet from the VL (a FIFO buffer) is sent to the forwarding table unit. Each entry in the forwarding table has a destination ID (called Destination Local Identifier—DLID) and a corresponding output port number. As per the IBA specification, we use a linear forwarding table implementation that is indexed by the destination ID in the header. The forwarding table provides the output port information to the arbiter (third stage), which resolves output port contentions.

WRR (Weighted Round-Robin) scheduling is used for selecting between VLs that are contending for the same output port, as shown in the figure.

In the last stage of the switch, the packets flowing out of the output ports of the crossbar are placed into the buffers that compose the output VLs. The packets from the output VLs are multiplexed onto a common output link using the IBA specified two-level VL arbitration. First, priorities between different VLs are determined by the priorities of SLs assigned to these VLs. Then, a WRR scheduling is used to schedule packets having the same SL.

2.2 NIC Architecture

Network interface cards (NICs), also known as Host Channel Adapters (HCAs) in the IBA terminology, are used for attaching processing nodes to a network. As shown in Fig. 2, a typical NIC consists of a processor to handle network traffic, a pair of DMA engines to handle data movement, and a local memory (typically DRAM) for buffers and doorbells. The $send/recv$ requests from the host are directly written on the memory mapped doorbell region. The NIC processor polls this region in a FIFO manner and programs the appropriate DMA engine(s) to process these requests. If data needs to be copied from (to) the host, "Host DMA Engines" are programmed or, if data needs be sent (received) to (from) the network, "Network DMA Engines" are used. We do not discuss the details of the Queue Pair (QP) structure of the HCA architecture since these are not included in our energy model. However, these are included in our simulator for enabling user-level communication. The details can be found in the IBA specification [15].

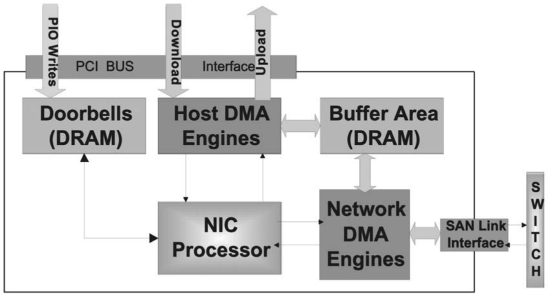


Fig. 2. A standard NIC architecture.

2.3 Link Architecture

The links are capable of sending 2.5Gbps data over lengths reasonable for cluster interconnects. The link includes the transmitter, receiver, and clock recovery at the receiver, as shown in Fig. 3. The link also supports multiple-frequency operation through the use of dynamic voltage scaling. In dynamic voltage scaling, the adaptive voltage/frequency unit (AV/AF unit) provides the minimum amount of voltage required to operate at a given frequency, while also providing the said frequency to the transmitter. As energy consumption scales as the square of voltage, this can result in significant energy savings. The link also supports a shutdown mode, where the transmitter, receiver, and adaptive supply unit are powered down completely, reducing energy consumption to nearly zero. Only a small detector in the receiver must remain powered in order to detect when the transmitter wishes to begin operation again. The 1V supply supports fast wake-up of the link and will be discussed in detail later.

3 Energy-Efficient System Design

In this section, we first provide an overview of the circuit design for the different components. These designs are used to obtain the power numbers used in the experiments and for further energy optimizations. Each of these components was layed out in a standard 180nm process and simulated in HSPICE to obtain energy consumption data. The main energy parameters are summarized in Table 4. We also used 70nm design to capture the impact of leakage energy with technology scaling. We then focus on various energy optimization techniques. Specifically, we explore the possibility of dynamically shutting down links, used in conjunction with an adaptive routing algorithm, and compare it with the dynamic voltage scaling technique for links. Since

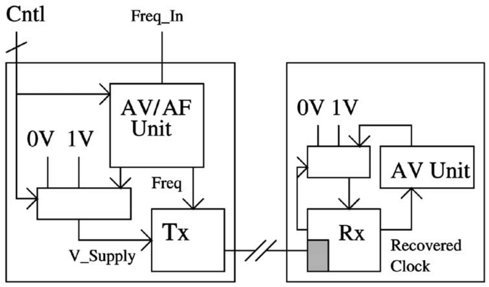


Fig. 3. Block diagram of a link.

the utility of the DVS scheme for links is already known [13], we will confine our focus to major DVS design issues in dynamic frequency/voltage change, transition overhead, and clock synchronization, which have not been addressed in detail previously.

3.1 Switch Design

The switch design corresponding to Fig. 1 focuses on four major components: the FIFO buffers, lookup tables, crossbar, and output port arbiters. The lookup tables are used to model the forwarding, WRR (to implement VL arbitration) and VL-SL mapping tables. Also, the FIFO buffer is used to model both the input and output VLs.

FIFO Buffer Design. The FIFO buffers are used in the input and output VLs. To implement the FIFO buffer, the head and tail pointer are implemented using two shift registers. A flipflop storing a bit of value one in the shift register identifies the head/tail location. The buffer design employs 7T SRAM cells with minimum sized transistors and uses bit line segmentation [19] to improve the performance and reduce energy consumption. The control logic is divided into segments as well and only segments that are near the current location of the pointer receive clock signals. This prevents unnecessary switching of cells that are not affected by the current operation.

Design Optimization. Our buffer design is performed in a leakage energy conscious fashion. While not a significant concern in 180nm technology, it will be an important factor in sub-100 nm technologies. We utilize the predictable access patterns of the FIFO buffers and the ground gating mechanism [20] to provide a leakage-energy optimized buffer design. In ground gating, an additional sleep transistor is placed between the memory cells of the buffer and the ground. When this transistor is turned on, the circuit operates normally. However, when it is off, the leakage current is significantly reduced, but the data is lost.

Our design breaks each buffer into a number of cells, where each cell has one sleep transistor. As the FIFO access pattern is deterministic, we power down cells after reading them since, by the nature of FIFO access, the data cannot be accessed again. Also, we can predict, without error, the minimum amount of time that must pass before a cell in sleep mode might be written to. Note that we may still incur the energy penalty of earlier activation, but our goal is to avoid introducing any additional performance penalties. The cell size is chosen such that, at maximum clock rate, the time required to traverse a cell is larger than the time required to power up the next cell. We thus power up a given cell n whenever the cell $(n-1)$ is first written to. This deterministic behavior of the FIFO buffers allows the supply gating to be implemented with a zero performance penalty. Fig. 4 shows the optimized buffer architecture. Table 1 shows the energy consumption of a quad-packet buffer designed in 70nm technology with different utilizations. We can save up to 80 percent of energy in the best case (no utilization) with this design.

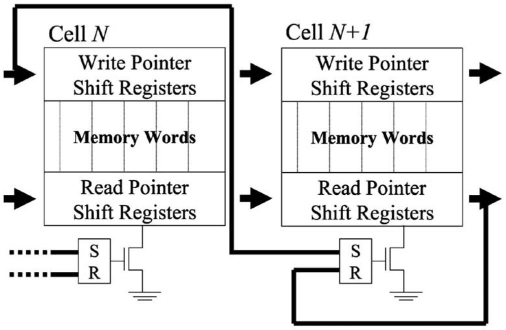

Cell N
Cell N+1
Write Pointer Shift Registers
Memory Words
Read Pointer Shift Registers
S
R
Write Pointer Shift Registers
Memory Words
Read Pointer Shift Registers
S
R

Fig. 4. Sleep mode buffer architecture.

The buffer design also plays an important role in supporting dynamic frequency scaling of links. We will discuss this issue later along with the link design.

Lookup Table Design. Tables in our switch design were all modeled as direct-mapped SRAMs and incorporated standard energy optimizations such word and bit line segmentation.

Crossbar Design. The interconnection fabric in the switch is an $n \times n$ crossbar, where n is the number of physical links. The crossbar design utilizes a nonblocking pass transistor-based design [21] and contains order n^2 pass transistors to connect any input line to another output. We capture the energy consumed in setting up the control signals of the crossbar as well as the actual data transfer from the input to the output.

Arbiter Design. Our arbitration circuit is an FCFS design. The FCFS arbiter requires $n \times C$ buffers, where n is number of physical ports and C is the number of VLs per physical link. In designing the buffers, all the energy optimizations techniques discussed for the I/O buffers were employed.

3.2 NIC Design

We modeled the main components of a typical NIC shown in Fig. 2, including a RISC processor, 8MB local memory, DMA controller, doorbell queue, and the VL arbiter (which is WRR in the HCA design). We use DRAM data sheets [22] to obtain the energy numbers for the local memory and the doorbell queue. To evaluate the energy consumed by the RISC in the NIC, we use a StrongARM 1100 RISC core [23] based energy simulator and execute the kernel code. Note

TABLE 1
Buffer Energy Consumption at Varying Utilizations
(70nm Design)

Buffer Utilization	0%	50%	100%
Dynamic Read Energy	8065pJ/packet		
Dynamic Write Energy	8408pJ/packet		
Static Power (Normal)	0.32W	0.32W	0.32W
Static Power (Optimized)	0.058W	0.19W	0.32W

that energy consumption in the processor could vary significantly based on the processor activity.

3.3 Link Design and Optimization

Our link uses a multiplexed serial link design [24]. The link uses differential current-mode signaling with an integrating receiver to reduce the impact of noise on the operation. In such a design, each bit is represented by the directional flow of current on the differential pair of wires and the current accumulates in capacative nodes to create the detected voltages. This design greatly reduces the impact of noise as the actual voltage of the lines does not matter, merely the relative amount of the current flowing through them. Clocking information is carried on a per-pin basis, inline with the data, by forcing a toggle on the differential pair at the desired clock frequency. This allows for high frequency operation with minimal jitter problems as the actual delay of each link is taken into account during bit reception. Each clock-toggle of the link is subdivided into five smaller time units at both the transmitter and receiver, allowing for a very high throughput as compared to the actual link-embedded clock cycle. The design also uses an adaptive voltage supply that minimizes the operating voltage at a given frequency to optimize the energy consumption. This link can be used in chip-to-chip PCB interconnects, as well as in short-length cable installs. The energy consumption per bit transmitted in the link is obtained by scaling the values reported for a 250nm link design that can be operated at different frequencies [25] to 180nm. Scaling was performed using rules identified in [26] and a single frequency link was designed in both 250nm and 180nm technologies, then simulated in HSPICE to validate the scaling used.

3.3.1 Dynamic Voltage Scaling (DVS)

We now discuss the first energy optimization in the links using DVS. Although the link design we are using supports multiple date rates and voltages, it was originally intended only to support static frequency operation from powerup. The frequency change at runtime can be accomplished by changing the input clock to the transmitter. In this scenario, the adaptive power supply will attempt to track the new frequency, while the receiver will attempt to regain lock to again receive data. Although the link design we are using supports multiple data rates and voltages, experimental data for frequency changes during operation was unavailable at the time of this writing. Therefore, we conservatively assume that the link cannot support data transfer during periods of frequency change. During the transition period, we conservatively utilize the higher of the two energy values consumed in the two transition states.

The time required for this transition is determined by two components of the transceiver, the clock-matching PLL and the adaptive supplies. PLL lock times on such a link are on the order of 400ns [27], while voltage lock times on the variable power supply can be much higher [24]. Assuming that the variable power supply has a tracking rate of 0.1V/μs shows that frequency transition in the link is limited by the power supply adjustment time in almost all cases. Table 2 shows the energy consumption of the scaled link, per bit, at various frequencies.

TABLE 2
Link Energy Consumption (180nm)

Rate(bps)	660M	995M	1.33G	1.66G	1.93G	2.31G	2.50G
pJ/bit	5.25	5.41	6.49	7.14	8.31	9.59	10.21

Clock Synchronization for DVS. A main issue with using DVS is the ability of the router to adapt itself to the new frequencies of the link and operate reliably. Synchronization between link and router clock frequencies is maintained by the I/O buffers. The buffers are designed using a ring-address pointer and a 7-T SRAM cell architecture. The ring-address system allows the buffer to operate using separate and independent read and write clock signals. The 7-T SRAM cell allows single-ended reads and writes to the cells, meaning both operations can be performed simultaneously without worry of conflict on the bit lines. As the two clock frequencies need not be even multiples of each other when employing DVS, it is possible that the two clocks will be operating with an unknown phase-shift between the two. During general case operation with the head and the tail pointers in separate locations in the memory, the two operations are independent, relying on different control circuitry and bit lines for operation. The two operational situations that could cause problems occur as the two address pointers approach each other in the memory. These situations correspond to buffer emptying and buffer full and prevention of synchronization problems is handled separately for each case.

The first case of buffer full occurs when the tail pointer approaches the head pointer. The buffer is divided into a number of cells of a given size (256 bits in our case) and each cell maintains a value known as *Active Data* on a latch. This latch is set whenever the tail pointer first enters a cell and is reset whenever the head pointer leaves the cell. The buffer circuitry raises the global buffer full signal anytime the tail pointer enters a cell preceding one that contains *Active Data*. The time for the *Buffer Full* signal to be raised is significantly smaller that the time for the next cell to be written, as shown in Fig. 5. The upper level circuitry recognizes this buffer full signal and stops attempting to write into the buffer at the end of the current cell. The distance between the tail and head pointer (cell size) that sends this buffer full signal can be varied at design time, allowing different transmit sizes such as packet switching or wormhole switching to be supported. This results in a slight buffer underutilization, but prevents all cases of the tail pointer reaching the head pointer.

The other case of operation, as the buffer empties, occurs as the read (head) pointer approaches the write (tail) pointer. In this case, errors are prevented by the action of the buffer empty signal. As the last read is performed, the empty signal propagates to the edge of the design in 550ps, much less time than it takes to generate the next read request. This prevents any more read requests from being generated until the empty signal is deactivated.

3.3.2 Dynamic Link Shutdown (DLS) Algorithm

We propose a second alternative to save energy by introducing a link shutdown mechanism, where a link can be powered down if it is not used heavily. Reenabling the link would normally incur a significant delay penalty while the link is powered up and reconfigured into the network. We describe two mechanisms to minimize this penalty. The

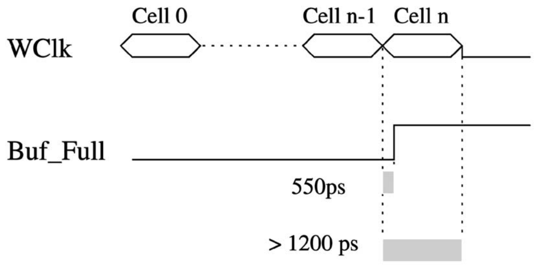


Fig. 5. Timing diagram of filling buffer.

first approach is simply to reduce the overhead at the circuit level and the second approach is to use alternative paths while a link is being powered up. While the forwarding of a packet along an existing operational (and, hence, loaded) link results in a slightly higher utilization for that link, the additional delay incurred due to said increased utilization is much smaller than that incurred by waiting for a link to power up again. In addition, forwarding along already operational links also helps in eliminating the overhead of global network reconfiguration by using the adaptivity information programmed in the local forwarding table.

The latency penalty incurred while the link powers up and regains timing lock can be minimized through the use of a multiplexed power supply, as shown in Fig. 3. While the adaptive supply regains lock, the transmitter is powered from the 1V multiplexed supply line. Normally, the adaptive supply would only supply 1V when the link was operating in the vicinity of 1Gbps, however, to prevent process variations and other variances from possibly causing a malfunction, we overdesign the supply. To power up the link, the transmitter begins sending control signals to the receiver, where a small circuit that remains powered at all times detects the modulation on the transmission channel and activates the receiver, also on a 1V multiplexed supply. The modulation detection circuit is little more than a comparator with hysteresis and a number of flip-flops. To initiate the wake-up procedure, the transmitter begins toggling the complementary data lines. The analog comparator will note the bit-inversion on the link and increment a counter, which decreases at a regular rate. If the bit-inversions occur frequently enough, the counter will reach a trigger-level and initiate wakeup in the receiver, including PLL lock to the incoming clock signal. Once the receiver has locked to a frequency, a response signal is sent, allowing valid data transmission to occur much sooner than if the normal adaptive supply had been used. While there exists a possibility that certain pathological noise patterns in the link could cause accidental wakeup, the unexpected receipt of this activation signal causes the transmitter to activate the link drivers, drowning out link noise. As such, erroneous or fictitious packets cannot be caused by accidental wakeup of the receiver. Once both the transmitter and receiver are activated, the link must operate at the minimum frequency of 640Mbps until the adaptive supply stabilizes. This allows the link to begin operation at 640Mbps much sooner than it would otherwise. The wake-up time for this situation is dominated by the lock-time of the receiver. Thus, we conservatively assume a wake-up time of 800ns, two PLL lock-times. As the only circuit activated in the power-down state is the modulation detector on the receiver, which consumes negligible power when compared to an active link, we assume that the link consumes no power when disabled.

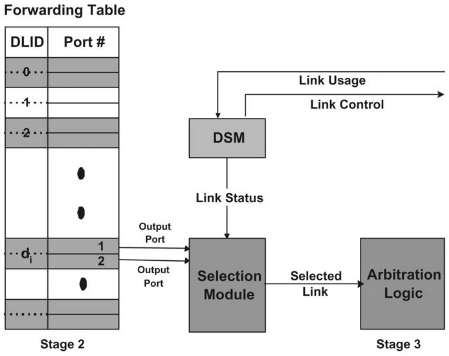


Fig. 6. Dynamic shutdown mechanism.

Even with the multiplexed power supply, we cannot avoid the delay incurred by reconfiguration overhead. Whenever the links are powered down/up, forwarding tables in all switches should be changed properly. For example, if we use the SPF (Shorted Path First) algorithm for the IBA-based SANs that has been proposed recently [16], each shutdown/up event needs to reexecute the algorithm to construct the appropriate forwarding tables. The global communication for this reconfiguration can be avoided by using a distributed adaptive routing scheme, which can provide alternate paths for a shutdown link. In this paper, we use a modified SPF algorithm for irregular networks and minimal adaptive routing for the 2D regular networks to provide alternate paths between a source and destination. This path is encoded in the forwarding table using the algorithms proposed in [16] for irregular networks. The size of this forwarding table is not constant for all nodes since it depends on the number of cycles containing the node. This table will not be changed due to link shutdowns and guarantees connectivity in the network as long as the shutdown links do not make the network disjoint. This is assured by the dynamic shutdown module (DSM) described next. Although IBA only permits the use of forwarding tables, where each entry has a destination ID and a corresponding output port number, it is possible to have several output ports for a destination by using multipath bits in the DLID, as shown in Fig. 6.

As shown in Fig. 6, we add two hardware modules for the proposed scheme: dynamic shutdown decision module (DSM) and outport selection module. These modules are placed between pipelines Stages 2 and 3 in Fig. 1. The shutdown decision module provides the link status to the selection module which finally selects one port out of the possible output ports encoded in the forwarding table. The link status is collected at a local node over a certain observation window (W_s). The selected port is sent to the arbitration logic. The DSM gets the usage information from Stage 5 of the switch and provides specific link control (shutdown/up) to the link control block shown in Fig. 3. This control signal is used to set the supply voltage of the link transmitter. The selection module uses the LFU (Least Frequently Used) policy for selecting one of the possible output ports.

The link shutdown decision is based on the concept that if we find a minimal set of links providing connectivity with minimum performance degradation, then a subset of the rest of links can be powered down. For this, we use two threshold utilization values ($T_{\max}$, T_I) to decide when we need to use all possible output ports for a destination and when we can shutdown a link due to its low utilization. Let D be the set of all destination nodes in the forwarding table of a node. For each destination node, we define its usage frequency as the sum of the utilization of all possible links that can be used for reaching the node. Our algorithm selects the set of nodes whose usage frequency is greater than T_{max}, which is denoted as D_a. All nodes in D_a may need to use all links since they are heavily used. Given D_a, the decision algorithm works as follows.

In the first case, when all nodes need to use all possible links ($D_a = D$), we can still shut down some links whose individual link utilization is less than the given threshold (T_I). In the second case ($D_a \neq D$), since some of the output port entries in the forwarding table have low link utilization (therefore, their total utilization is less than $T_{\max}$), we can shut down some links. Let L_{high} be the set of links for D_a and these links will be active. Initially, L_s, the set of candidate links to be shutdown is equal to $(L - L_{\text{high}})$. Then, we find the nodes that cannot be reached using only L_{high}. Let D_{on} be the set of nodes which can be routed with only links in L_{high}. Then, D_{off} $(= D - D_{\text{on}})$, are the set of nodes that cannot be routed with links in L_{high}. If $D_{\text{off}} = \emptyset$, all nodes can be routed using L_{high} links and, thus, we can shutdown L_s. Otherwise, we need to find the minimal set of links which provides connectivity. For this, we first find a link that provides maximum connectivity ($l_{\max}$) by counting the number of appearances in the forwarding table. Then, D_{on} becomes the set of nodes that can be routed with $l_{\max}$ among the D_{off} nodes. D_{off} and L_s are then updated as ($D_{\text{off}} - D_{\text{on}}$) and ($L_s - \{l_{\max}\}$), respectively. These updates of D_{on} and D_{off} are conducted recursively until $D_{\text{off}} = \emptyset$. Finally, the links in L_s can be shut down. The decision algorithm is summarized below.

```
(1) $D_a = \{d \mid \sum_{l \in L_d} U_l > T_{\text{max}}\}$, where $D$ is the total
    set of destination nodes, $L$ is the total set of links, $L_d$
    is the set of links for a node $d$ provided by Stage 2, and
    $U_l$ is the usage count of link $l$.
(2) If $D_a = D$, shutdown $\exists\ l,\ U_l < T_I$.
(3) Else
    $L_{\text{high}} = \cup_{d \in D_a} d,\ \ D_{\text{on}} = \{d \mid l \in L_d,\ l \in L_{\text{high}}\}$,
    $D_{\text{off}} = D - D_{\text{on}}$, and $L_s = L - L_{\text{high}}$.
     (a) If $D_{\text{off}} = \emptyset$, shutdown $L_s$.
     (b) Else repeat
           find a link ($l_{\text{max}}$) in $D_{\text{off}}$ that provides
           maximum connectivity.
           Update $D_{\text{on}}$ ($D_{\text{on}} = \{d \mid l_{max} \in L_d\ ,\ d \in D_{\text{off}}\}$ )
           $D_{\text{off}} = D_{\text{off}} - D_{\text{on}}$ and $L_s = L_s - \{l_{\text{max}}\}$.
         Until $D_{\text{off}} = \emptyset$.
```

Shutdown Decision Algorithm

4 Experimental Platform

We have developed a simulation testbed for the cluster interconnect that models switches, HCAs, and links conforming to the IBA specification. The simulation models bit level activity and provides cycle accurate information for both performance and energy estimates.

In addition to the timing simulation, we incorporate energy numbers extracted from actual layouts of the components described in the prior section. Each of the components of the system discussed in the previous section were custom designed and simulated using HSPICE in 180nm technology using a supply voltage of 1.8V to extract the power consumption values. Then, we redesigned the switch components and a single frequency link using 70nm technology. These energy numbers are used along with the activity monitored in the different components of the cluster interconnect to derive the energy consumption results.

Our simulation model is flexible in that one can specify the number of physical links, number of VLs per physical link, link bandwidth, packet size, and many other architectural and power parameters. The power numbers presented in this work use actual values from design and circuit simulation. Our simulator is also flexible in supporting different network topologies. To illustrate this flexibility, we simulate a 15-node irregular network and an (8×8) 2D mesh network designed using 5-port switches. For the 15-node irregular networks, we use SPF routing [16], while, for the mesh network, we use minimal adaptive routing to support routing adaptivity. We simulated both packet switched and wormhole switched networks.

For the experiments, the best-effort traffic is generated with a given injection rate λ and follows the Poisson distribution. The size of best-effort packets is assumed to be fixed and a destination is picked using a uniform distribution. These assumptions provide the most general case of a network analysis. We then use ON/OFF traffic [28] to generate traffic burst and hot spot distribution of destinations to examine the energy impact of the proposed techniques. For ON/OFF traffic, during the OFF period, the source does not generate any messages, while, during the ON period, messages are generated according to the given injection rate λ_{onoff}.

For some of the experiments, the workload includes packets from real-time VBR traffic, which is generated as a stream of packets between a pair of communicating (source-destination) processors. The traffic in each stream is generated from seven MPEG-2 traces [29], where each trace has a different bandwidth requirement. Important statistics of the different traces are shown in Table 3. Each stream generates 30 frames/sec and each frame is divided into fixed-size packets, where each packet consists of the MTU and the header.

The important output parameters measured in our experiments are average packet latency (includes network latency and source queuing delay in HCA) and energy consumption in microjoules. For real-time VBR traffic, Deadline Missing Time (DMT) is measured whenever a frame misses its deadline. Table 4 summarizes the main parameters used in our experiments. Note that leakage energy at 180nm is not significant and, as such, has been omitted from this table. For link energy consumption, we use data from Table 2.

5 Experimental Results

We start our discussion by comparing the energy and performance behavior of the link optimization schemes. The distribution of energy consumption, shown in Fig. 7a, indicates the importance of focusing on the energy consumption of the links. For each injection rate, the six bars indicate the following combinations from left to right:

TABLE 3
MPEG-2 Video Sequence Statistics (Kbits)

Video Seqs	Avg Bandwidth Reqs (Kb/s)	Avg Sz of I Frame	Avg Sz of P Frame	Avg Sz of B Frame
1	7,138.2	430.2	295.6	194.2
2	15,231.4	839.4	680.6	401.0
3	13,526.4	534.7	569.4	387.8
4	8,536.5	393.3	340.3	241.4
5	6,124.2	350.6	242.9	172.9
6	18,406.0	974.7	721.9	529.6
7	13,497.9	637.6	536.0	394.3

MPEG streams have I, P, and B frames.

TABLE 4
Energy Consumption and Default System Configuration Parameters with 180nm

Component	Status	Energy (pJ)
Quad-Packet Buffer (per packet-size operation)	Read	16431
	Write	14298
Lookup Table	Active	310.00
Arbitration	Active	6.10086
Crossbar (per packet)	Active	2739
DRAM in HCA (per 1 packet)	R/W	3570
RISC processor in HCA (average per cycle at low load)		108

General System Parameters		
T_{max}/T_I	40%	3%
Window Size : DVS (W_d), DLS (W_s)	150 cycles	300 cycles
Physical Link Bandwidth		2.5 Gbps
Number of Physical Links		5
Number of VLs/Physical Link		16
Header Size (LRH, BTH, and CRC fields)		26 bytes
Maximum Transfer Unit (MTU)		1024 bytes
Input/Output VL Buffer Size		4200 bytes

1. (No DVS, No Shutdown, No Adaptive Routing),
2. (DVS, No Shutdown, No Adaptive Routing),
3. (No DVS, No Shutdown, Adaptive Routing),
4. (No DVS, Shutdown, Adaptive Routing),
5. (DVS, No Shutdown, Adaptive Routing), and
6. (DVS, Shutdown, Adaptive Routing).

Thus, we observe that the link energy optimization schemes have a significant influence on overall energy savings. With optimized links, the energy consumptions for the HCAs and the switches will become more important since the energy consumption in these two components almost doubles as the injection rate increases from 20 percent to 60 percent. This phenomenon happens because the unoptimized links consume the same amount of energy, regardless of whether they transmit data and, hence, are not influenced by the injection rate variation. On the contrary, HCA and switch energy consumption depends on the injection rate variation. As the memory elements dominate the energy consumption of the HCAs and the switches, the increasing importance of the leakage energy will make the energy optimizations for these parts crucial as well.

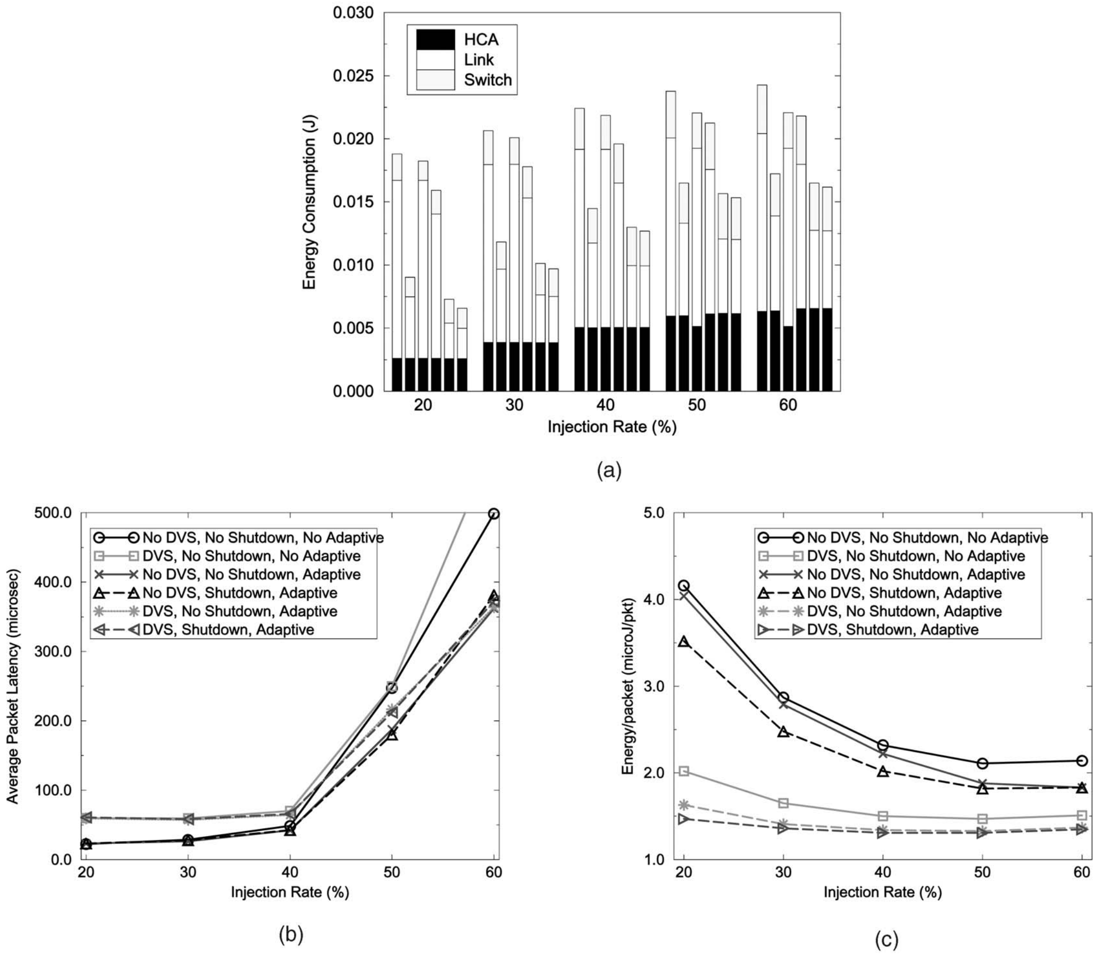
In Fig. 7b, we observe that the use of only DVS (Case 2) increases latency over the entire workload. Specifically, as compared to the base case (Case 1), the latency increased by 500 percent at a 20 percent injection rate and, at a 60 percent injection rate, the latency is increased by 10 percent. The combination of adaptive routing and DVS (Case 5) can mitigate this performance penalty at only high load since adaptivity has little impact at low load. In contrast, the DLS scheme (Case 4) has almost the same latency (only 3 percent degradation) as compared to Case 3. The combination of both DVS and DLS (Case 6) still has high latency at lower load, but approaches the best case at 60 percent load.

The energy behavior in Fig. 7c reveals that it is more energy efficient to operate at a higher injection rate than at a lower injection rate. This is contrary to the trend in network latency in which lower injection rates are preferred as they result in lower latency. Further, the DVS scheme that performed poorly when considering latency reduces the energy required per packet by half at low loads. It must be noted that most of the energy savings occur from the voltage scaling in the links. However, DVS also increases the amount of time the packets spend in the buffers (in the switches) before being delivered to the destination. The increased buffer utilization by itself does not increase the energy consumption in the 180nm technology. However, as leakage energy in buffers becomes significant in designs using sub-100nm technology, the increased time spent in the buffers can become an energy concern, as will be shown later.

The DLS scheme provides a more moderate energy saving as compared to DVS (in Fig. 7c) as only a small percentage of links can be shut down completely. However, when combined with DVS, it can provide an additional 50 percent savings. This additional savings results from the fact that about half of the links that would have operated at the lowest frequency in DVS can be completely shut down with support from adaptive routing. The distribution of frequencies (voltages) at which the links operate (shown in Fig. 10) helps to identify the source for the additional savings. The percentage of links that can be shut down decreases by 10 percent as the load increases from 20 percent to 60 percent. The additional savings provided by shutdown scheme decrease in a similar fashion as the load increases.

Fig. 8 shows the DMT of MPEG-2 video traffic when we change the number of VLs in a 15-node irregular network with DVS and DLS. In this figure, we can observe that the DMT increases dramatically beyond 60-70 percent. Therefore, Fig. 7 and Fig. 8 show that the operating range of between 40-60 percent load provides an opportunity to strike an appropriate balance between energy consumption and performance constraints.

The arrival process was then changed to use an On/Off source to generate traffic burst and message destinations were generated using a hot spot model. This provides a better stress testing of the network compared to the general model used in Fig. 7. The impact of the different schemes is plotted in Fig. 9. With this workload, voltage transition happens more often in DVS and DLS, resulting in higher penalties in the average packet latency. However, we obtain significant energy savings with the optimized schemes (Fig. 9b) and the relative contribution of DLS with respect to DVS is more pronounced (see Fig. 7b and Fig. 9b). We also experiment with these schemes in a large 2D network. The first three upper lines on the left side represent the energy per packet and the others represent the average packet



(a)

(b)

(c)

Fig. 7. 15-Node irregular network results (180nm design). (a) Energy consumption. (b) Average package latency. (c) Energy/packet.

latency for the base scheme without any optimization, DVS scheme, and DLS scheme, respectively. Strikingly, we find that the energy savings due to shutdown is 38 percent at low load, while the performance degradation is limited to 4 percent. From these results, we can conclude that the energy saving increases, while the performance penalty remains almost constant as the network size increases.

As technology scaling results in increased leakage energy, we have simulated a 70nm design where leakage energy is significantly higher. The optimized buffer is used for the higher leakage results as an example of how to mitigate the effects of this leakage increase. We experiment with the impact of DVS and DLS using these different leakage parameters to see the effect of technology evolution on these schemes. The results, shown in Fig. 11, indicate that DVS provides better energy optimization than the DLS scheme with current technology, but has the maximum latency. This high latency implies that the packets spend more time in the buffers and, thus, the buffer utilization increases. When technology scales and leakage current becomes a dominant factor, DLS becomes better than DVS at high load due to the lower buffer space utilization in DLS. This allows more of the buffer to stay powered down, reducing the leakage penalty. This crossover occurs between 40 percent and 50 percent workload. (At 50 percent load, the average number of powered cells with DLS scheme is 35,087 and with DVS scheme is 56,749.)

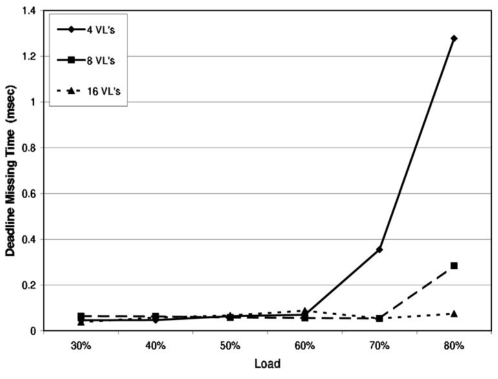
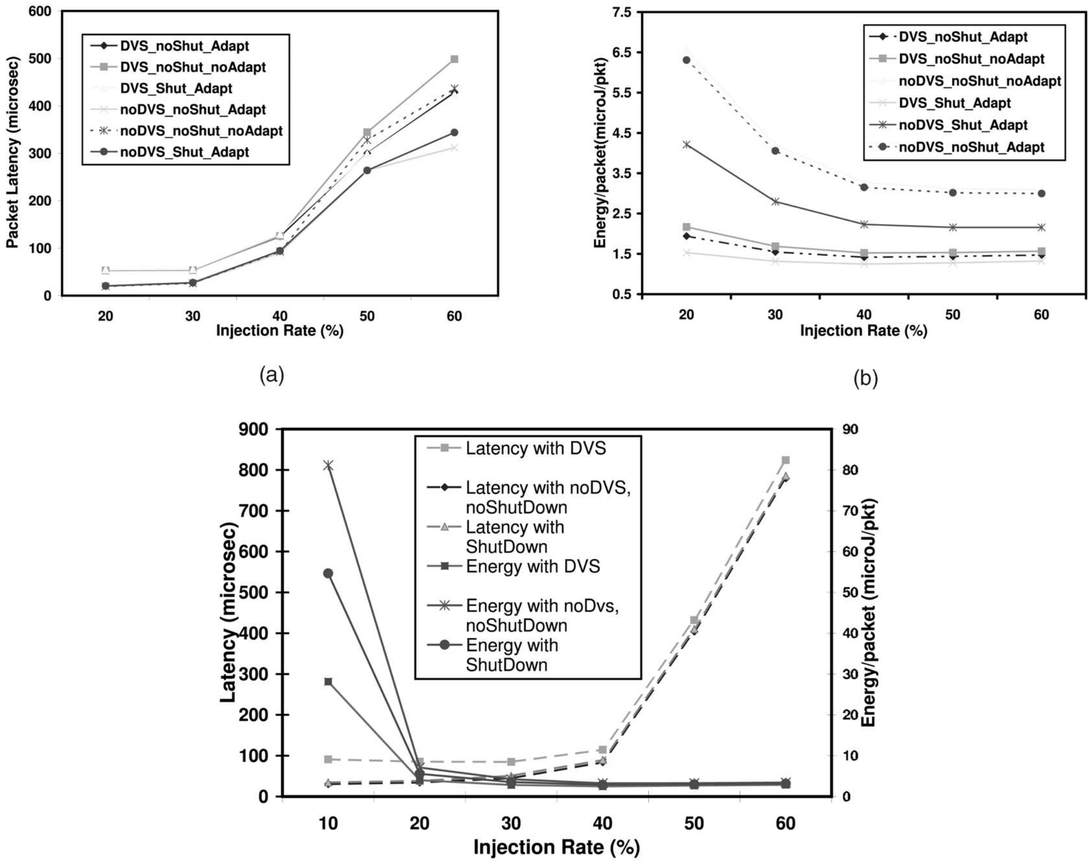


Fig. 8. Deadline missing time of MPEG-2 video traffic in a 15-node network.

In the previous experiments, we used 180nm technology, where the buffer leakage energy is negligible. In Fig. 12, we investigate the performance results of packet-switched and wormhole-switched (8×8) mesh networks in a single frequency 70nm design. As shown in Fig. 12a, a wormhole network outperforms a packet-switched network even with smaller buffers in terms of average packet latency. A wormhole network with 4-packet length buffers shows better performance than packet-switched networks with 4-packet length buffers and 8-packet length buffers. Fig. 12b



(a)

(b)

(c)

Fig. 9. Results with traffic burst and nonuniform destination distribution (180nm design). (a) Average packet latency. (b) Energy/packet. (c) 8×8 network results.

shows the two switching networks with the same buffer size (4-packet) with/without optimized buffer. While the packet-switched network suffers from higher latency, when combined with the optimized buffer it consumes the least energy/packet. The wormhole network with the optimized buffer dissipates almost the same energy up to a load of 40 percent, but the advantage of the optimization disappears as the load increases. Fig. 12c shows the energy dissipation by the links and the switches. The HCA energy is not included here since we do not have the model for the NIC processor and DRAM at 70nm design. For each injection rate, the four bars indicate the packet switch with the optimized buffers (Case 1), the packet switch with the unoptimized buffers (Case 2), the wormhole switch with optimized buffers (Case 3), and the wormhole switch with the unoptimized buffers (Case 4). With unoptimized buffers (Case 2 and Case 4), the distribution of energy dissipation is roughly 50 percent for both links and the switches, while it was 80 percent for the links and 20 percent for the switches with 180nm technology, as shown in Case 1 of Fig. 7a. This

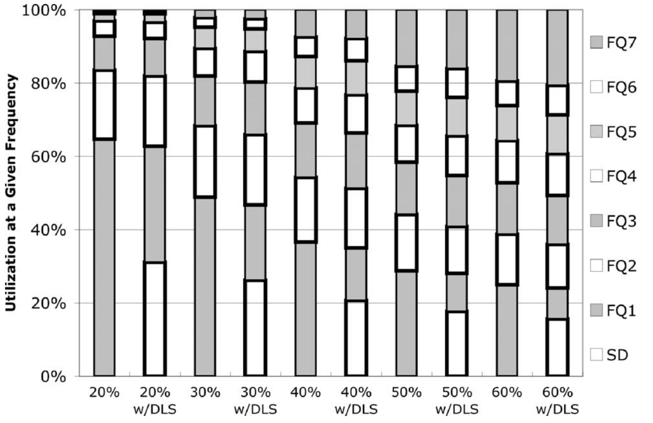


Fig. 10. Frequency profiling with both DVS and DLS.

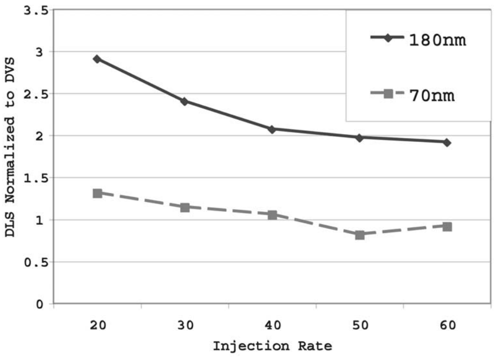
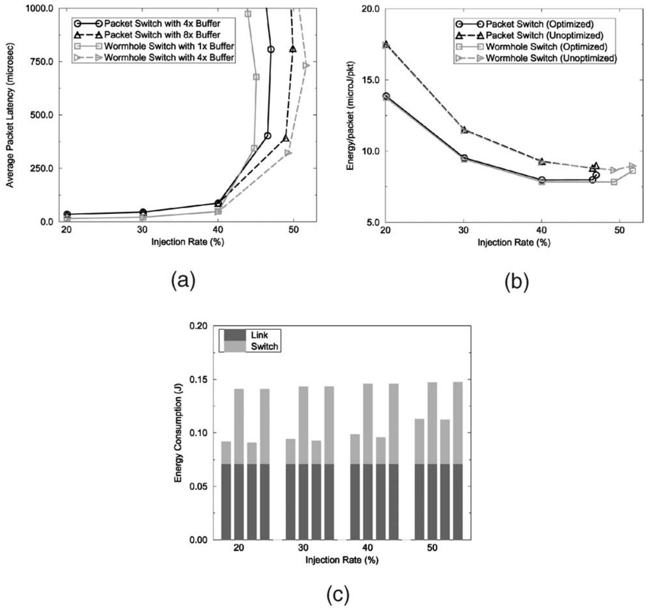


Fig. 11. Comparison of energy consumed by DLS and DVS as technology scales (normalized to DVS).



(a)

(b)

(c)

Fig. 12. 8×8 mesh network results (70nm design). (a) Average packet latency. (b) Energy/packet. (c) Energy distribution.

is because, as technology evolves, the memory elements dominate the energy consumption. With the buffer optimization scheme, which reduces the leakage energy, we can save more than 60 percent energy in buffers at low load.

6 Conclusions

Along with performance, energy consumption is becoming an important concern for designing cost-effective clusters that are emerging as the dominant mode of computing infrastructure. Unlike processor and memory designs, energy-efficient interconnects for SANs have received little attention and the recently released IBA specification clearly documents the commercial interest in this direction.

In this paper, we have presented a holistic approach for energy characterization and optimization of an IBA style cluster interconnect. We have designed an integrated simulation tool that combines the architecture level design artifacts with energy parameters from actual designs of the switch, NIC, and links for providing various performance and energy estimates. Since the links and buffers consume a major portion of the total energy budget, our investigation focused on optimizing energy consumption in these two components. In this context, we presented a detailed design and analysis of DVS and a new dynamic link shutdown technique (DLS) for the links and an optimized buffer design to reduce leakage energy.

The main conclusions of this work are the following: First, while DVS is a viable technique to reduce power consumption in the links, it degrades network latency significantly at low to medium load. Second, DVS and the buffer energy consumption need to be examined together as buffer utilization increases due to DVS. This becomes especially critical at 70nm technology, where buffer leakage energy is a dominant factor. Our study reveals that energy saving with DVS suffers due to leakage current and becomes more prominent as the load increases beyond 40 percent. Third, the proposed DLS technique is an elegant and feasible approach to optimize both performance and power. It can provide up to 40 percent energy savings with less than 5 percent performance penalty. DLS can be nicely blended with a suitable adaptive routing algorithm for intelligent path selection so that underutilized links can be powered down without incurring high performance penalty. Finally, integration of DVS and DLS provides the best energy optimization.

Currently, we are pursuing several extensions of this research. The link model will be extended to optical links. The memory and processor power consumption in the NIC can be optimized with better designs. The network energy analysis with real workloads should provide more meaningful results. Finally, we plan to conduct an exhaustive energy characterization of a complete cluster system.

Acknowledgments

This research was supported in part by US National Science Foundation grants NSF-0202007, NSF-0208734, NSF-0429631, and CAREER-0093085, as well as the MARCO/DARPA GSRC Power-Aware Systems Awards. A preliminary version of this work appeared in the *Proceedings of the International Symposium on Low-Power Electronics and Design*, 2003.

References

[1] *The New York Times,* “There’s Money in Housing Internet Servers,” Apr. 2001, http://www.internetweek.com/story/INW20010427S0010.

[2] R. Bryce, “Power Struggle,” Dec. 2000, http://www.zdnet.com.au/newstech/enterprise/story/0,2000025001,20107749,00.htm.

[3] Mellanox Technologies Inc., “Mellanox Performance, Price, Power, Volumn Metric (PPPV),” http://www.mellanox.com/products/shared/PPPV.pdf, 2004.

[4] Digital Equipment Corp., *Alpha Architecture Technical Summary,* 1992.

[5] W. Dally, P. Carvey, and L. Dennison, “The Avici Terabit Switch/Router,” *Proc. Hot Interconnects 6,* pp. 41-50, Aug. 1998.

[6] T.T. Ye, L. Benini, and G.D. Micheli, “Analysis of Power Consumption on Switch Fabrics in Network Routers,” *Proc. 39th Conf. Design Automation (DAC),* pp. 524-529, June 2002.

[7] G.-Y. Wei, M. Horowitz, and A. Kim, “Energy-Efficient Design of High-Speed Links,” *Power Aware Design Methodologies,* M. Pedram and J. Rabaey, eds., chapter 8, Kluwer Academic, 2002.

[8] G.-Y. Wei, J. Kim, D. Liu, S. Sidiropoulos, and M. Horowitz, “A Variable-Frequency Parallel I/O Interface with Adaptive Power-Supply Regulation,” *IEEE J. Solid-State Circuits,* vol. 35, pp. 1600-1610, Nov. 2000.

[9] C.S. Patel, S.M. Chai, S. Yalamanchili, and D.E. Schimmel, “Power Constrained Design of Multiprocessor Interconnection Networks,” *Proc. Int’l Conf. Computer Design (ICCD),* pp. 408-416, Oct. 1997.

[10] A.G. Wassal and M.A. Hasan, “Low-Power System-Level Design of VLSI Packet Switching Fabrics,” *IEEE Trans. Computer-Aided Design of Integrated Circuits and Systems,* vol. 20, pp. 723-738, June 2001.

[11] H.-S. Wang, L.-S. Peh, and S. Malik, “A Power Model for Routers: Modeling Alpha 21364 and InfiniBand Routers,” *Proc. Hot Interconnects 10,* pp. 21-27, Aug. 2002.

[12] H.-S. Wang, X. Zhu, L.-S. Peh, and S. Malik, “Orion: A Power-Performance Simulator for Interconnection Networks,” *Proc. 35th Int’l Symp. Microarchitecture (MICRO),* pp. 294-305, Nov. 2002.

[13] L. Shang, L.-S. Peh, and N.K. Jha, “Dynamic Voltage Scaling with Links for Power Optimization of Interconnection Networks,” *Proc. IEEE Int’l Symp. High-Performance Computer Architecture (HPCA),* pp. 91-102, Feb. 2003.

[14] V. Soteriou and L.-S. Peh, “Dynamic Power Management for Power Optimization of Interconnection Networks Using On/Off Links,” *Proc. 11th Symp. High Performance Interconnects (Hot Interconnects),* Aug. 2003.

[15] InfiniBand Trade Assoc., “InfiniBand Architecture Specification, Volume 1, Release 1.0,” Oct. 2000, http://www.infinibandta.org.

[16] E.J. Kim, K.H. Yum, C.R. Das, M. Yousif, and J. Duato, "Performance Enhancement Techniques for InfiniBand™ Architecture," *Proc. IEEE Int'l Symp. High-Performance Computer Architecture (HPCA),* pp. 253-262. Feb. 2003.

[17] J. Duato, S. Yalamanchili, and L. Ni, *Interconnection Networks: An Engineering Approach.* San Francisco: Morgan Kaufmann, July 2002.

[18] W.J. Dally, "Virtual-Channel Flow Control," *IEEE Trans. Parallel and Distributed Systems,* vol. 3, pp. 194-205, May 1992.

[19] M.B. Kamble and K. Ghose, "Analytical Energy Dissipation Models for Low-Power Caches," *Proc. 1997 Int'l Symp. Low Power Electronics and Design (ISLPED),* pp. 143-148, Aug. 1997.

[20] M.D. Powell, S. Yang, B. Falsafi, K. Roy, and T.N. Vijaykumar, "Reducing Leakage in a High-Performance Deep-Submicron Instruction Cache," *IEEE Trans. Very Large Scale Integration (VLSI) Systems,* vol. 9, pp. 77-90, Feb. 2001.

[21] E. Geethanjali, V. Narayanan, and M.J. Irwin, "An Analytical Power Estimation Model for Crossbar Interconnects," *Proc. IEEE Int'l ASIC/SOC Conf.,* pp. 119-123, Sept. 2002.

[22] T.G. Tip, "RDRAM Power Estimation and Thermal Considerations," Oct. 2001, http://www.rambus.com/rdf/presentations/2_A3_Thermal_Yip2.pdf.

[23] T. Simunic, L. Benini, and G.D. Micheli, "Cycle-Accurate Simulation of Energy Consumption in Embedded Systems," *Proc. 36th Conf. Design Automation (DAC),* pp. 867-872, June 1999.

[24] J. Kim and M. Horowitz, "Adaptive Supply Serial Links with Sub-1V Operation and Per-Pin Clock Recovery," *Proc. Int'l Solid-State Circuits Conf. (ISSCC),* pp. 268-269, Feb. 2002.

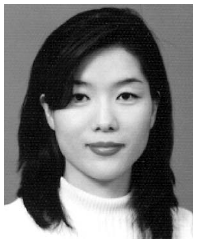
[25] J. Kim and M. Horowitz, "An Efficient Digital Sliding Controller for Adaptive Power Supply Regulation," *IEEE J. Solid-State Circuits,* vol. 37, pp. 639-647, May 2002.

[26] S. Borkar, "Design Challenges of Technology Scaling," *IEEE Micro,* vol. 19, pp. 23-29, July-Aug. 1999.

[27] D. Duarte, Y.-F. Tsai, N. Vijaykrishnan, and M.J. Irwin, "Evaluating Run-Time Techniques for Leakage Power Reduction," *Proc. ASP-DAC/VLSI Design,* pp. 31-38, Jan. 2002.

[28] J. Qiu and E. Knightly, "QoS Control via Robust Envelop-Based MBAC," *Proc. Sixth Int'l Workshop Quality of Service (IWQoS '98),* pp. 62-64, May 1998.

[29] M.B. Caminero, F.J. Quiles, J. Duato, D.S. Love, and S. Yalamanchili, "Performance Evaluation of the Multimedia Router with MPEG-2 Video Traffic," *Proc. Third Int'l Workshop Comm., Architecture, and Applications on Network Based Parallel Computing (CANPC '99),* pp. 62-76, Jan. 1999.



Eun Jung Kim received the BS degree in computer science from the Korea Advanced Institute of Science and Technology, Korea, in 1989, the MS degree in computer science from Pohang University of Science and Technology, Korea, in 1994, and the PhD degree in computer science and engineering from Pennsylvania State University in 2003. From 1994 to 1997, she worked as a member of the technical staff in the Korea Telecom Research and Development Group. She is currently an assistant professor in the Department of Computer Science at Texas A&M University. Her research interests include computer architecture, parallel/distributed systems, computer networks, cluster computing, QoS support in cluster networks and Internet, performance evaluation, and fault-tolerant computing. She is a member of the IEEE, IEEE Computer Society, and ACM.

Greg M. Link received the BE degree in electrical engineering (2002) from Pennsylvania State University and the BS degree in physics (2002) from Juniata College. He is currently working toward the PhD degree in computer science and engineering at Pennsylvania State University under Dr. N. Vijaykrishnan. His areas of interest include network-on-chip design, thermal-aware computing, and application-specific processors. He is a student member of the IEEE.

Ki Hwan Yum received the BS degree in mathematics from Seoul National University, Korea, in 1989, the MS degree in computer science and engineering from Pohang University of Science and Technology, Korea, in 1994, and the PhD degree in computer science and engineering from the Pennsylvania State University in 2002. From 1994 to 1997, he was a member of the technical staff in the Korea Telecom Research and Development Group. He is currently an assistant professor in the Department of Computer Science at the University of Texas at San Antonio. His research interests include computer architecture, parallel/distributed systems, cluster computing, and performance evaluation. He is a member of the IEEE, IEEE Computer Society, and ACM.

N. Vijaykrishnan received the BE degree in computer science and engineering from SVCE, University of Madras in 1993 and the PhD degree in computer science and engineering from the University of South Florida, Tampa, in 1998. Since 1998, he has been with the Computer Science and Engineering Department at Pennsylvania State University, where he is currently an associate professor. His research interests are in the areas of energy-aware reliable systems, embedded Java, nano/VLSI systems, and computer architecture. He has authored more than 100 papers in these areas. He serves as the vice-chair for Student Activities for the IEEE Computer Society. He is a member of the IEEE and the IEEE Computer Society,

Mahmut Kandemir received the BSc and MSc degrees in control and computer engineering from Istanbul Technical University, Istanbul, Turkey, in 1988 and 1992, respectively. He received the PhD degree from Syracuse University, Syracuse, New York, in electrical engineering and computer science in 1999. He is an associate professor in the Computer Science and Engineering Department at the Pennsylvania State University. His main research interests are optimizing compilers, I/O intensive applications, and power-aware computing. He is a member of the IEEE, IEEE Computer Society, and ACM.

Mary J. Irwin received the PhD degree in computer science from the University of Illinois in 1977. She has been on the faculty at the Pennsylvania State University since 1977, where she currently holds the title of the A. Robert Noll Chair in Engineering in the Department of Computer Science and Engineering. Her research and teaching interests include computer architecture, embedded and mobile computing systems design, power aware design, and electronic design automation. She is a fellow of the IEEE, a fellow of the ACM, and was elected to the National Academy of Engineering (NAE) in 2003. She currently serves as a member of the Technical Advisory Board of the Army Research Lab, on ACM's Publications Board, and as the Editor-in-Chief of ACM's *Journal on Emerging Technologies in Computing Systems* (*JETC*).

Chita R. Das received the MSc degree in electrical engineering from the Regional Engineering College, Rourkela, India, in 1981 and the PhD degree in computer science from the Center for Advanced Computer Studies, University of Louisiana, Lafayette, in 1986. Since 1986, he has been with the Pennsylvania State University, where he is currently a professor in the Department of Computer Science and Engineering. His main areas of interest are parallel and distributed computing, cluster computing, mobile computing, Internet QoS, multimedia systems, performance evaluation, and fault-tolerant computing. He is currently an associate editor of the *IEEE Transactions on Computers* and has served on the editorial board of the *IEEE Transactions on Parallel and Distributed Systems.* He is a fellow of the IEEE and a member of the ACM.