Energy Optimization Techniques in Cluster Interconnects*

E. J. Kim K. H. Yum‡ G. M. Link N. Vijaykrishnan
M. Kandemir M. J. Irwin M. Yousif† C. R. Das

Department of Computer Science and Engineering
Pennsylvania State University
University Park, PA 16802
ejkim,link,das,vijay,kandemir,mji@cse.psu.edu

‡Department of Computer Science
The University of Texas at San Antonio
San Antonio, TX 78249
yum@cs.utsa.edu

ABSTRACT

Designing energy-efficient clusters has recently become an important concern to make these systems economically attractive for many applications. Since the links and switch buffers consume the major portion of the power budget of the cluster, the focus of this paper is to optimize the energy consumption in these two components. To minimize power in the links, we propose a novel dynamic link shutdown (DLS) technique. The DLS technique makes use of an appropriate adaptive routing algorithm to shutdown the links intelligently. We also present an optimized buffer design for reducing leakage energy. Our analysis on different networks using a complete system simulator reveals that the proposed DLS technique can provide optimized performance-energy behavior (up to 40% energy savings with less than 5% performance degradation in the best case) for the cluster interconnects.

Categories and Subject Descriptors

C.2.1 [**Computer-Communication Networks**]: Network Architecture and Design

General Terms

Design, Performance

Keywords

Buffer Design, Cluster Interconnect, Dynamic Voltage Scaling, Dynamic Link Shutdown, Energy Optimization, Link Design, Switch Design

1. INTRODUCTION

Wide spread use of cluster systems in a diverse set of applications has spurred significant interest in designing such servers considering performance, scalability, and quality-of-service (QoS) as the design objectives. In addition to these parameters, optimizing 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 [10]. The energy consumption is critical as it affects the cost of cooling and backup power generation. In fact, new data centers in New York area are forecast to increase the city's power demands by 25% [10]. Recent technology trend 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.

The interconnection fabric consumes a significant portion of the total cluster power. For example, the integrated switch in the Alpha 21364 is reported to consume 20% of the chip budget, while 33% of the router linecard power is consumed in the interconnect in the Avici switch [1]. Similarly, the routers and links in a Mellanox server blade, consume almost the same power as that of a processor (15W) and is about 37% of the total power budget. These numbers indicate that power dissipation in the interconnect is significant and needs careful investigation.

Only 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 [14]. Techniques for optimizing power dissipation in high speed links have been proposed in [13]. Analytical power models for interconnection networks have been developed based on transistor counts [6]. Wang et al. have presented an analytical power model to explore different switch configurations [12]. Recently, Shang et al. extended the dynamic voltage scaling (DVS) technique to optimize link power in regular interconnection networks [8]. It was shown that link DVS can conserve significant link energy at the expense of network performance.

We present a new link energy optimization technique called dynamic link shutdown (DLS). The proposed DLS scheme 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. In order to benefit from DLS, we present an adaptive routing strategy that intelligently uses a subset of links for communication, thereby facilitating dynamic link shutdowns for minimizing energy consumptions. In this paper, we compare our scheme to the existing DVS approach. Our evaluation shows that DVS incurs a high performance penalty in low to medium workloads, though it results in significant power savings. Specifically, the average network latency degradation varied from 500% to 10% as the network load changed from 20% to 60%. On the other hand, the proposed DLS technique can provide moderate energy saving with minimal degradation in average network latency. Further, we observe that advantage of DVS diminishes as leakage energy becomes more important with technology scaling to 70nm. The high network latency associated with DVS results in increased buffer utilization thereby increasing the overall leakage energy consumption. Finally, we show that integration of both DVS and DLS results in the best energy optimization.

The organization of the paper is as follows: in Section 2, the system architecture is discussed. The proposed link energy

*This research was supported in part by NSF grants CCR-9900701, CCR-0098149, CCR-0208734, NSF CAREER 0093085, and a Grant from GSRC.

†Dr. Yousif is with Advanced Component Division, Intel Coporation. E-mail: mazin.s.yousif@intel.com

ISLPED'03, August 25–27, 2003, Seoul, Korea.

optimization technique is presented in Section 3. In Section 4, the experimental platform and the simulation results are discussed, followed by the concluding remarks in Section 5.

2. SYSTEM ARCHITECTURE

We have developed a simulation testbed for the cluster interconnect that models switches, NICs and links conforming to the Infiniband (IBA) specification [3]. The simulation models the flow of the flits at the bit level across the different fabrics and tracks the activity of each of the components every cycle. In addition to the timing simulation, we incorporate energy numbers extracted from actual layouts of the switch components. Each of the major components of the switch 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 the 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. In this section, we describe the switch fabric, NIC, and link architectures that were designed.

2.1 Switch

Architecture: The n-port switch modeled adopts a five-stage pipelined packet-switched model, as shown in Figure 1. The model can be easily changed to capture wormhole switching or virtual cut-through switching. The pipelined model represents the recent trend in router design [1]. Our IBA compliant switch design supports virtual lanes (VLs) that provide a mechanism to implement multiple logical flows over a single physical link. The IBA specification allows between 2 and 16 VLs.

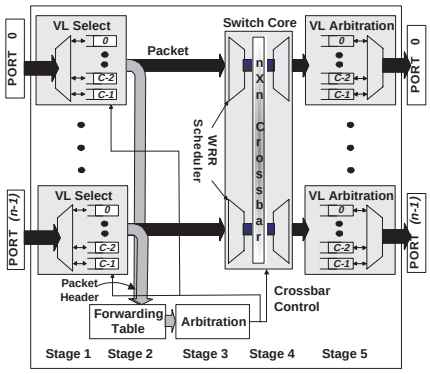


Figure 1: A Pipelined Switch Architecture

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 header of a packet from the VL (a FIFO buffer) is sent to the forwarding table. 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. To select one of the contending VLs for the same crossbar input port, we use WRR (Weighted Round Robin) scheduling.

In the last stage of the switch, the packets flowing out of the output ports of the crossbar are buffered in 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.

Energy Modeling: The switch design involved design of four components: FIFO buffers, lookup tables, crossbar and output port arbiter. The lookup tables are used to model the forwarding, WRR (to implement VL arbitration) and VL-SL mapping tables. The main energy parameters obtained from the layouts are summarized in Table 3. We also used 70nm design to capture the impact of leakage energy as technology scales. Hence, the buffer design used to obtain the energy consumption data is performed in a leakage energy conscious fashion. Specifically, we utilize the predictable access patterns of the FIFO buffers and the ground gating mechanism [7] 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. When it is off, the leakage current is significantly reduced and the data is lost.

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

Table 1: Buffer Energy Consumption at Varying Utilizations. (70nm design)

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 the data is not needed 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. Table 1 shows the energy consumption of a quad-packet buffer designed in 70nm technology with different utilizations. We can save up to 80% of energy in the best case (no utilization) with this design.

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 Figure 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 used, or if data needs be sent(received) to(from) the network, "Network DMA Engines" are used.

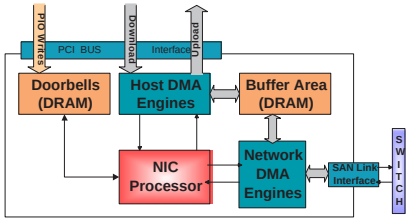


Figure 2: A Standard NIC Architecture

Energy Modeling: We modeled the main components of a typical NIC shown in Figure 2, including a RISC processor, 8MB local memory, DMA controller, doorbell queue, and the VL arbiter. We use DRAM data sheets [11] 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 [9] based energy simulator and execute the kernel code.

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

Table 2: Link Energy Consumption (180nm)

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 Figure 3. The link also supports multiple-frequency operation through the use of DVS. In DVS, the adaptive voltage/frequency unit (AV/AF unit) provides the minimum voltage required to operate at a given frequency, while also providing the said frequency to the transmitter. The link also supports a shutdown mode, where the transmitter, receiver, and adaptive supply unit are powered down completely, reducing energy consumption to near zero. Only a small detector in the receiver must remain powered, in order to detect when the transmitter wishes to begin operation again. The optimization and modeling of the link are the focus of the next section.

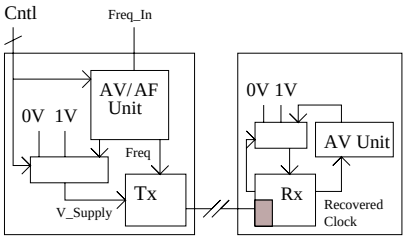


Figure 3: Block Diagram of a Link

3. LINK ENERGY AND OPTIMIZATION

Our link is based on the multiplexed serial link design [5]. This link uses 5:1 multiplexing and an adaptive voltage supply that minimizes the operating voltage at a given frequency to optimize the energy consumption.

3.1 Dynamic Voltage Scaling (DVS)

Although the link design we are using supports multiple data rates and voltages, experimental data for frequency change during operation was unavailable at the time of this writing. The frequency change at run-time is 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. Therefore, the link does not 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 [2], while voltage lock times on the variable power supply can be much higher [5]. Using a power supply with a relatively fast 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 seven frequency settings.

3.2 Dynamic Link Shutdown (DLS) Algorithm

An alternative to save link energy is to add hardware such that a link can be powered down if it is not used heavily. Re-enabling 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 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. The latter also helps in eliminating the overhead of global network reconfiguration by using the adaptivity information programmed in the local forwarding table.

The penalty on link powerup can be avoided through the use of a multiplexed power supply, as shown in Figure 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 over-design the supply. To powerup 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. 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. Once 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, and is equal to 800ns. As the only circuit activated in the power-down state is the modulation detector on the receiver, we assume negligible energy consumption during the power-down state.

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. For example, if we use the SPF (Shortest Path First) algorithm for the IBA-based SANs that has been proposed recently [4], each shutdown/up event needs to re-execute 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 provides alternate paths for a shutdown link. In this paper, we use a modified SPF algorithm for irregular networks and the X-Y routing for the 2-D regular networks to provide alternate paths between a source and destination. This path is encoded in the forwarding table. We do not provide further details of the SPF algorithm. 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, it is possible to have several output ports for a destination by using multipath bits in the DLID as shown in Figure 4.

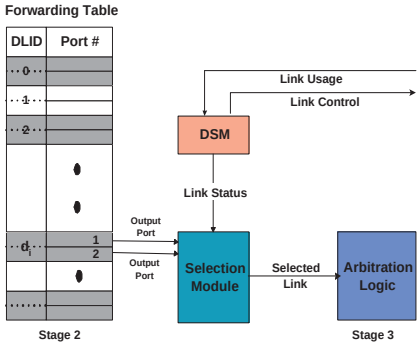


Figure 4: Dynamic Shutdown Mechanism

As shown in Figure 4, we add two hardware modules for the proposed scheme: a dynamic shutdown decision module (DSM) and an outport selection module. These modules are placed between pipelines Stages 2 and 3 in Figure 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 Figure 3. This control signal is used to set the supply voltage of the link transmitter. The selection module uses 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 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 shutdown 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 shutdown 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 shutdown. The decision algorithm is summarized below.

(1) $D_a = \{d \mid \sum_{l \in L_d} U_l > T_{\max}\}$, 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 ($l_{\max}$) in D_{off} that provides maximum connectivity.
Update D_{on} ($D_{on} = \{d | l_{max} \in L_d\ ,\ d \in D_{off}\}$)
$D_{\text{off}} = D_{\text{off}} - D_{\text{on}}$ and $L_s = L_s - \{l_{\max}\}$.
Until $D_{\text{off}} = \emptyset$.

Shutdown Decision Algorithm

4. EXPERIMENTAL PLATFORM AND RESULTS

Our simulation model allows the user to specify the number of physical links, number of VLs per physical link, link bandwidth, packet size, and many other architectural and power parameters. Currently, the power numbers embed actual values from design and circuit simulation. Our simulator supports different network topologies. To illustrate this feature, we simulate a 15-node irregular network and an (8×8) 2-D mesh network designed using 5-port switches. For the 15-node irregular networks, we use the SPF routing while for the mesh network, we use the (X-Y) routing to support routing adaptivity. We simulated both packet switched and wormhole switched networks.

Component	Status	Energy(pJ)
Quad-Packet Buffer(per packet)	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
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

Table 3: Default System Configuration Parameters with 180nm

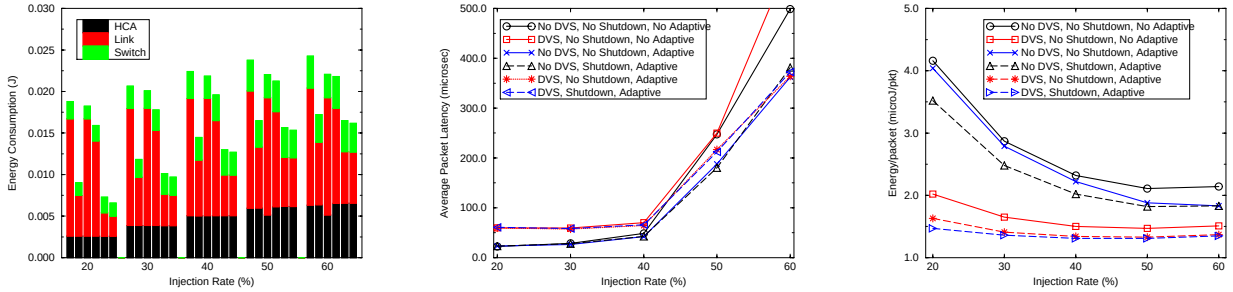
For most of the experiments, traffic is generated with a given injection rate λ, and follows the Poisson distribution. A destination is picked using a uniform distribution. These assumptions provide the most general case of a network analysis. We then use ON/OFF traffic to generate traffic burst and hot spot distribution of destinations to examine the energy impact of the proposed techniques.

The important output parameters measured in our experiments are average packet latency (includes network latency and source queueing delay in HCA), and energy consumption in microjoules. Table 3 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.

We start our discussion by comparing the energy and performance behavior of the link optimization schemes. The distribution of energy consumption, shown in Figure 5(a), 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; (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% to 60%. 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 Figure 5 (b), we observe that the use of only DVS (case 2) increases latency over the entire workload. Specifically, the latency is increased by 500% at the injection rate of 20% and by 10% at 60% injection rate, as compared to the base case(case 1). 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% 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% load.

The energy behavior in Figure 5 (c) 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. 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



(a) Energy Consumption (b) Average Packet Latency (c) Energy/packet

Figure 5: 15-Node Irregular Network Results (180nm design)

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 Figure 5 (c)) as only a small percentage of links can be shutdown completely. However, when combined with DVS it can provide an additional 50% 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 shutdown with support from adaptive routing. The distribution of frequencies (voltages) at which the links operate (shown in Figure 9) helps to identify the source for the additional savings. The percentage of links that can be shutdown decreases by 10% as the load increases from 20% to 60%. The additional savings provided by shutdown scheme decrease in a similar fashion as the load increases.

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 Figure 5. The impact of the different schemes are plotted in Figure 6. 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 (Figure 6(b)), and the relative contribution of DLS with respect to DVS is more pronounced (See Figure 5(b) and Figure 6(b)). We also experiment with these schemes in a large 2-D network. The first three upper lines on the left side represent the energy per packet, and the others represent the average packet latency for the base scheme without any optimization, DVS scheme, and DLS scheme, respectively. Strikingly, we find that the energy saving due to shutdown is 38% at low load, while the performance degradation is limited to 4%[1]. 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 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 the 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% and 50%

[1]We observe that by tuning the parameters (T_{max} and T_I) in DLS, we can control the energy and performance. However, we show only the results with the parameters that provide similar performance with the base scheme due to space limitations.

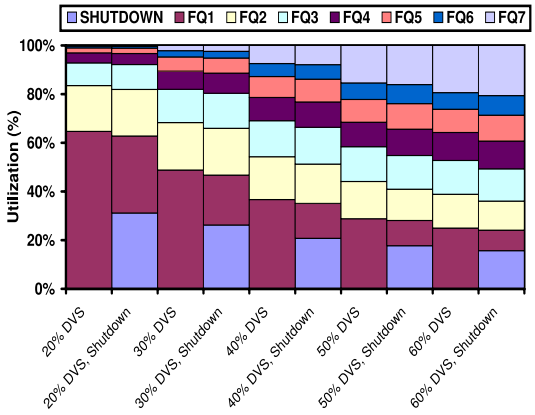


Figure 7: Frequency Profiling with Both DVS and DLS

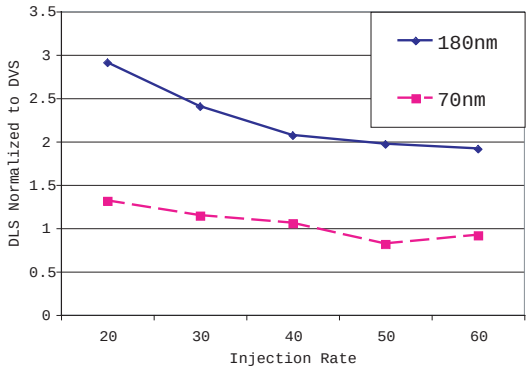
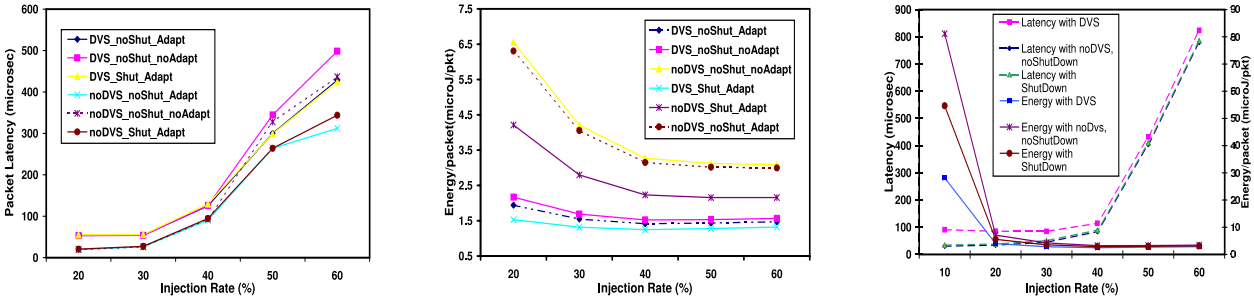


Figure 8: Energy Consumption with Technology Scaling

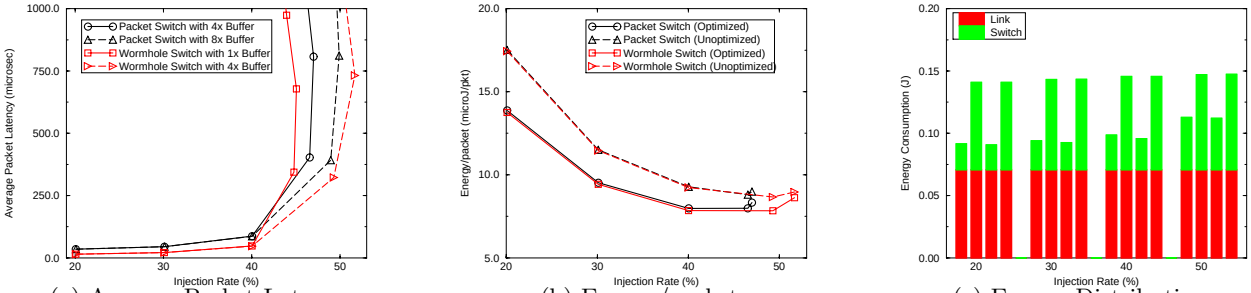
workload. (At 50% load, the average number of powered cells with DLS scheme is 35087 and with DVS scheme is 56749.)

In Figure 9, we investigate the performance results of packet-switched and wormhole-switched (8×8) mesh networks in a single frequency 70nm design. As shown in Figure 9 (a), 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. Figure 9 (b) 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%, but the advantage of the optimization disappears as the load increases. Figure 9 (c) 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



(a) Average Packet Latency (b) Energy/packet (c) 8×8 Regular Network Results

Figure 6: Results with Traffic Burst and Nonuniform Destination Distribution (180nm design)



(a) Average Packet Latency (b) Energy/packet (c) Energy Distribution

Figure 9: 8×8 Mesh Network Results (70nm design)

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% for both links and the switches, while it was 80% for the links and 20% for the switches with 180nm technology, as shown in case 1 of Figure 5(a). This 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% energy in buffers at low load.

5. CONCLUSIONS

In this work, we focused on the energy estimation and optimization of the cluster intereconnects. We can draw the following conclusions from this work: 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%. Third, the proposed DLS technique is an elegant and feasible approach to optimize both performance and power. It can provide up to 40% energy savings with less than 5% performance penalty. DLS can be nicely blended with a suitable adaptive routing algorithm for intelligent path selection so that under-utilized links can be powered down without incurring high performance penalty. Finally, integration of DVS and DLS provides the best energy optimization.

6. REFERENCES

[1] W. Dally, P. Carvey, and L. Dennison. The Avici Terabit Switch/Router. In *Proc. Hot Interconnects 6*, 1998.

[2] D. Duarte, Y.-F. Tsai, N. Vijaykrishnan, and M. J. Irwin. Evaluating Run-Time Techniques for Leakage Power Reduction. In *Proc. ASP-DAC*, pages 31–38, 2002.

[3] InfiniBand Trade Association. InfiniBand Architecture Specification, Volume 1, Release 1.0, October 2000. Available from `http://www.infinibandta.org`.

[4] E. J. Kim, K. H. Yum, C. R. Das, M. Yousif, and J. Duato. Performance Enhancement Techniques for InfiniBand Architecture. In *Proc. HPCA*, February 2003.

[5] J. Kim and M. Horowitz. Adaptive Supply Serial Links with Sub-1V Operation and Per-Pin Clock Recovery. In *Proc. ISSCC*, 2002.

[6] C. S. Patel, S. M. Chai, S. Yalamanchili, and D. E. Schimmel. Power Constrained Design of Multiprocessor Interconnection Networks. In *Proc. ICCD*, 1997.

[7] 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. on VLSI*, 9, February 2002.

[8] L. Shang, L.-S. Peh, and N. K. Jha. Dynamic Voltage Scaling with Links for Power Optimization of Interconnection Networks. In *Proc. HPCA*, February 2003.

[9] T. Simunic, L. Benini, and G. D. Micheli. Cycle-Accurate Simulation of Energy Consumption in Embedded Systems. In *Proc. DAC*, 1999.

[10] The New York Times. There's money in housing internet servers, April 2001. `http://www.internetweek.com/story/INW20010427S0010`.

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

[12] H.-S. Wang, X. Zhu, L.-S. Peh, and S. Malik. Orion: A Power-Performance Simulator for Interconnection Networks. In *Proc. MICRO*, November 2002.

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

[14] T. T. Ye, L. Benini, and G. D. Micheli. Analysis of Power Consumption on Switch Fabrics in Network Routers. In *Proc. DAC*, 2002.