

BEST EFFORT MINIMAL ROUTING FOR FLY-OVER: A LIGHT WEIGHT DISTRIBUTED
MECHANISM FOR ENERGY EFFICIENT NETWORK-ON-CHIP

A Thesis

by

SHILPA BHOSEKAR

Submitted to the Office of Graduate and Professional Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of
MASTER OF SCIENCE

Chair of Committee,	Eun Jung Kim
Co-Chair of Committee,	Jiang Hu
Committee Member,	Weiping Shi
Head of Department,	Miroslav. M. Begovic

August 2018

Major Subject: Computer Engineering

Copyright 2018 Shilpa Bhosekar

ABSTRACT

Scalable Networks-on-Chip (NoCs) have become the *de facto* interconnection mechanism in large scale Chip Multiprocessors. NoCs devour a large fraction of the on-chip power budget of which static NoC power consumption is becoming the dominant component as technology scales down. Hence, reducing static NoC power consumption is critical for energy-efficient computing. Previous research suggests power-gating routers attached to inactive cores so as to save static power, but requires centralized control and global network knowledge. Moreover, packet deliveries in irregular power-gated network suffer from detour or waiting time overhead to either route around or wake up off routers. *Fly-Over (FLOV)* is a distributed power-gating mechanism to minimize static power consumption in NoCs without the need for global network information. However, the existing FLOV routing algorithm introduces unnecessary detours and pressurizes the routers in *AON* column resulting in high packet latencies and network congestion.

This work proposes FLOV+, *Best-Effort Minimal Routing Algorithm for Fly-Over (FLOV)* to route the packets using the shortest path in an irregular power-gated network and also relieve the stress on the *AON* column. This routing algorithm aims to minimize the average packet latency and to sustain throughput in network with power-gated routers. Synthetic workload evaluations show that the proposed algorithm reduces average packet latency upto 9.84% in an 8-dimensional mesh network. Simulation results also show 50% and 40% improvement in the network throughput for restricted FLOV and generalized FLOV power gating mechanisms respectively.

DEDICATION

To Aai and Baba, for their unconditional love and support

ACKNOWLEDGMENTS

I would like to express gratitude to my committee chair, Dr. E J Kim; Co-Chair, Dr. Jiang Hu and my committee members, Dr. Weiping Shi, for their guidance and support throughout the course of this research.

I would also like to express my sincere thanks to the guidance extended by all the members of the HPC Lab with the Computer Science Department at Texas A&M University, particularly to Dr. Kim and Jiayi Huang. Thanks also go to my department faculty and staff for making my time at Texas A&M University a great experience. Lastly, but with utmost importance, I am grateful to the support extended by my friends and family throughout the course of Master's degree.

CONTRIBUTORS AND FUNDING SOURCES

Contributors

This work was supervised by my thesis committee consisting of Dr. Eun Jung Kim of the Computer Science Department and Dr. Jiang Hu and Dr. Weiping Shi from Electrical Department at Texas A&M University.

Boyapati et al. have proposed *Fly-Over*, a distributed power gating mechanism to minimize static power consumption in routers used in NoCs. It was presented at International Parallel and Distributed Processing Symposium (IPDPS) in 2017. Current work is an extension to the research carried out on FLOV by Dr. Rahul Boyapati and Jiayi Huang, from the HPC Lab with the Computer Science Department at Texas A&M University under the guidance of Dr. E J Kim.

All other work conducted for the thesis was completed independently by the student.

Funding Sources

There are no outside funding contributions to acknowledge related to the research and compilation of this document

NOMENCLATURE

AON	Always ON
CCL	Credit Control Logic
CMP	Chip Multi Processor
CPU	Central Processing Unit
DSENT	Design Space Exploration for Network Tool
FLOV	Fly-Over
FM	Fabric Manager
FSM	Finite State Machine
gFLOV	Generalized Fly-Over
gFLOV+	Generalized Fly-Over with <i>Best-Effort Minimal Routing</i>
HSC	Hand Shake Control
LT	Link Traversal
NoC	Network-on-Chip
NoRD	Node Router Decoupling
PSR	Power State Register
OS	Operating Systems
rFLOV	Restricted Fly-Over
rFLOVopt	Restricted Fly-Over with <i>Best-Effort Minimal Routing</i>
RC	Route Computing
RP	Router-Parking
RPA	Router-Parking Aggressive
RPC	Router-Parking Conservative

SoC	System on Chip
ST	Switch Traversal
VASA	VC Allocation and Speculative Switch Allocation
VC	Virtual Channel

TABLE OF CONTENTS

	Page
ABSTRACT	ii
DEDICATION	iii
ACKNOWLEDGMENTS	iv
CONTRIBUTORS AND FUNDING SOURCES	v
NOMENCLATURE	vi
TABLE OF CONTENTS	viii
LIST OF FIGURES	x
LIST OF TABLES.....	xi
1. INTRODUCTION.....	1
2. BACKGROUND	4
2.1 NoC Architecture	4
2.1.1 Network Interface	4
2.1.2 Routers	4
2.1.3 Global Network Links.....	6
2.2 Performance and Cost.....	6
3. RELATED WORK	8
3.1 Router Parking	9
3.2 FLY-OVER	9
3.2.1 FLOV Router Architecture.....	10
3.2.2 FLOV Handshake Protocols	11
3.2.3 Dynamic Routing Algorithm.....	12
4. BEST-EFFORT MINIMAL ROUTING ALGORITHM FOR FLY-OVER.....	14
4.1 Motivation	14
4.2 Key Idea	14
5. EXPERIMENTAL EVALUATION.....	18

5.1 Simulator Configuration	18
6. RESULTS AND ANALYSIS	21
6.1 Latency Analysis	21
6.2 Power Consumption.....	27
6.3 Throughput and Scalability Analysis.....	28
7. CONCLUSION.....	30
REFERENCES	31

LIST OF FIGURES

FIGURE	Page
2.1 NoC and Baseline NoC Router Architecture.	6
4.1 Routing Algorithm Examples: X indicates a power-gated router, Example 1 uses FLOV Routing, Example 2 and 3 use <i>Best-Effort Minimal Routing</i>	15
6.1 Average Latency and Power Comparison for Injection Rates of 0.02(left) and 0.08(right) flits/node/cycle with Uniform Random Traffic.....	22
6.2 Packet Latency Breakdown For Uniform Random Traffic with Injection Rate of 0.02 (a) and 0.08 (b) flits/cycle/core.	24
6.3 Average Latency and Power Comparison for Injection Rates of 0.02(left) and 0.08(right) flits/node/cycle with Tornado Traffic.	25
6.4 Packet Latency Breakdown For Tornado Traffic with Injection Rate of 0.02 (a) and 0.08 (b) flits/cycle/core.	26
6.5 Static Power Comparison of FLOV with RP and Baseline.	27
6.6 Load-Latency Curves for Uniform Random Traffic with 50% Cores Power-Gated for 6×6 (a), 8×8 (b) and 10×10 Mesh Network	29

LIST OF TABLES

TABLE	Page
5.1 Simulation Testbed Parameters	19

1. INTRODUCTION

Recent years have seen a rise in multi-core architectures owing to increasing power consumption and diminishing returns in the performance of uniprocessor architectures. This multi-core wave has led to integrating hundreds and even thousands of cores on a single chip. Moore's law [1], makes this possible as shrinking transistor sizes has allowed denser on-chip packaging. CMPs extract maximum performance gains using parallel programming paradigms. Performance gains in CMPs not only depend on effectively utilizing all the cores but also on efficient data communication among multiple CMP components. Thus, the interconnect fabric connecting the CMP components must be scalable and should aim to provide low latency, high throughput and low power consumption.

Traditionally, shared bus architectures have been used as interconnection networks in CMPs due to their low-cost and simple control characteristics. A bus is a collection of wires connecting various on-chip components to facilitate data communication. However, with the higher degree of integration of on-chip components, data communication gets slowed down because of the long distance between components and large capacitive load on the bus. Moreover, shared bus architectures use arbitration logic to resolve contention among bus request accesses from multiple components. Arbitration logic serializes these multiple bus access requests causing severe communication delays. As a result, performance gains from shared buses do not scale well with increasing core count, typically greater than 10. Similarly, area and power requirements of crossbars restrict their usage as interconnect networks for large scale CMPs.

NoCs have emerged as a promising alternative to manage on-chip traffic efficiently in large scale multi-core systems. NoC architecture is heavily inspired from distributed computing networks. In NoC based architecture, each core is connected to a router and the routers are connected through point-to-point connections similar to a grid. NoCs employ packet-switch communication, where packets travel along the data links and the routers take care of buffering, and routing the packet in the right direction to reach its destination in the network. A high level of parallelism

can be achieved using NoCs as network links in NoC operate simultaneously on different data packets. In NoCs, dynamic power is dissipated for both communicating data across links as well as for switching and storage within the routers [2], while static power is consumed when routers and links are powered-on even when they are not used. Recent studies [3, 4, 5] have also shown that NoCs consume a significant portion of about 10% to 36% of the total on-chip power budget. Thus, developing power-efficient NoC designs is of the highest priority for power-constrained future CMPs.

Static power consumption of the on-chip circuitry is increasing at an alarming rate with the scaling down of feature sizes and chip operating voltages towards near-threshold levels. Previous studies [6, 7, 8, 9, 10] have shown that the percentage of static power in the total NoC power consumption increases from 17.9% at 65nm, to 35.4% at 45nm, to 47.7% at 32nm and to 74% at 22nm. As per this trend, for sub-10nm feature sizes, static power will become the major portion of the NoC power consumption. Furthermore, the failure of Dennard Scaling [11] has led to a dark silicon era, where certain portion of the chip needs to be turned off to meet power constraints. Thus, future CMP designs will have to work under stricter power envelopes where all the components on the chip cannot be run simultaneously without breaking the power and thermal constraints.

Power gating is a widely used technique to mitigate the worsening impact of static power consumption. Power gating turns off the power supply to the idle components of the system preventing leakage current flow, thereby saving up on static power consumption. A study by the International Data Corporation (IDC) [12] has suggested that servers achieve utilization of only 10% to 15%. Low core utilization indicates that routers attached to inactive cores are also idle for most of the time as the core doesn't inject or eject traffic into or from the network respectively. They are most likely involved in routing the traffic between the active cores. Research [13] has shown that on-chip routers can be idle 30%~70% of the time depending on the physical location of the routers in the NoC and work load. Therefore, power-gating techniques can be applied to on-chip routers taking advantage of their idle time. However, it is a challenging task to power-gate the routers because a) power gating the routers could incur additional packet latency due to detours, b) inter-

mittent packet arrivals might cut short the long period of router idle time required to compensate for the power gating overhead, reducing the efficiency of power-gating techniques and c) network might get partitioned, breaking the network connectivity. NoC router power-gating mechanisms, therefore, needs to address the above mentioned challenges.

Boyapati et al. propose Fly-Over (FLOV) [14], a light-weight power gating mechanism, that tries to power gate the routers as soon as the attached cores are powered down by the OS, in a distributed manner. Since, such a distributed power-gating mechanism may create interconnect partitions without communication paths, FLOV links are provided in power-gated routers to enable incoming packets to travel straight through them for maintaining the network connectivity. This way, routers can be gated for longer durations of time, compensating for the power gating overhead.

However, the existing FLOV routing algorithm introduces unnecessary detours and pressurizes the routers in MC column resulting in high packet latencies and network congestion. This work proposes FLOV+ , *Best-Effort Minimal Routing Algorithm* for FLOV, an improved version of the existing routing algorithm. It aims to minimize the packet detours or average packet latency and sustain throughput in a network with power-gated routers.

2. BACKGROUND

NoC is an emerging paradigm for communications in large scale multi-core systems. NoCs provide scalable and high bandwidth communication for large scale CMPs. It facilitates data transfer among multiple components such as processor cores, memories, specialized IP blocks present in the CMP system. It provides point-to-point data links that route packets to their destinations with the help of routers. This chapter briefly explains general NoC architecture and the functionality of the building blocks involved in designing NoCs.

2.1 NoC Architecture

A NoC system consists of three main building blocks, namely Network Interface(NI), routers and global network links.

2.1.1 Network Interface

NI establishes connection between IP cores and the on-chip network. NI injects packets from the core into the network and ejects data packets from the network to the core. As different components in the CMP follow different protocols, NI brings uniformity in the on-chip network as it packs and unpacks data as per the communication protocol required by the network. It also allows separation between computation and communication in the CMP system.

2.1.2 Routers

As the message is injected into the network, it is broken down into packets. Packets are further divided into fixed length flow control units called flits. Each packet consists of one head flit, multiple body flits and one tail flit. The head flit holds the destination information. Rest of the flits follow the same path as the head flit. The NoC architecture is governed by four parameters, namely topology, routing, flow control and router microarchitecture. Effect of each of these parameters in managing on-chip traffic is discussed below.

- *Topology*: Topology determines the physical layout and connections between nodes and channels(or links) in the network. Topology governs the number of hops required for the message traversal and has a direct impact on packet latency. Topology's effect on hop count directly impacts the energy consumption as message traversing through links and routers consume power. This work uses a mesh based topology. Mesh is a k -ary n -cube, where k is number of nodes along each dimension and n is the number of dimensions. In each dimension, k nodes are connected to their nearest neighbors. 8-ary 2-cube network configuration is used in this work.
- *Routing Algorithm*: For a given topology, routing algorithm decides the path to be taken by the packets to reach the destination. Ability of the routing algorithm to balance network traffic directly impacts the packet latency and throughput of the network. This work proposes *Best Effort Minimal Routing Algorithm, (FLOV+)* for *FLOV* as a part of this work aiming to improve the performance gains while mitigating the drawbacks of the existing *FLOV* routing algorithm.
- *Flow Control*: Flow control manages resource allocation, i.e. buffers and channel bandwidth as the message travels along the network. In this work, NoC is configured to use *Credit based Wormhole switching* flow control mechanism. Wormhole switching manages resources at the granularity of flits. With wormhole switching, a flit is allowed to move to the next router on the way as soon as sufficient buffer is available in the downstream router to hold this flit. Credit based flow control manages the buffer availability by the number of credits available at the downstream router.
- *Baseline Router Architecture*: The baseline microarchitecture used in this work is based on a state-of-the-art 3-stage virtual-channel router [15]. Figure 2.1 shows the main building blocks of the baseline router: input buffers, routing computation logic, VC allocator, switch allocator, and crossbar. The processing inside router is pipelined into 3 stages: Routing Computation (RC), VC Allocation and speculative Switch Allocation (VASA), and finally

Switch Traversal (ST). The output port to which a packet should traverse is computed in the RC stage based on the destination information in the head flit. In the VASA stage, an available VC in the next downstream router is assigned to this packet based on the credit information. At the same time, speculative arbitration between the inputs and outputs of the crossbar is processed in parallel. The flits with an assigned VC and the successfully granted switch will traverse the crossbar in the ST stage. Finally, Link Traversal (LT) is external to the router pipeline and is also assumed to take one clock cycle.

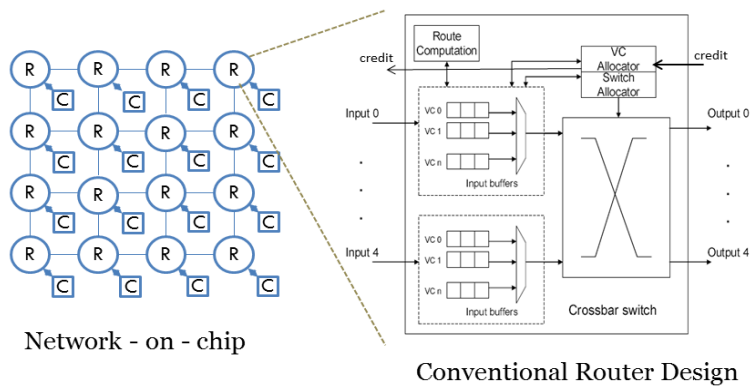


Figure 2.1: NoC and Baseline NoC Router Architecture.

2.1.3 Global Network Links

A communication link is a set of wires connecting two routers in the network. NoC link has two physical channels enabling full duplex communication between the routers. Channel width, the number of wires in the channel, is maintained constant throughout the network. In most cases, a flit corresponds to a *phit*, the physical unit, that is the least amount of data that can be transmitted on the link at once. Flit width is generally equal to the channel width.

2.2 Performance and Cost

Performance of on-chip networks is measured in terms of network latency or throughput. Latency denotes the time required by the message to travel from source to the destination node. Latency is calculated as the number of hops traversed times the latency to traverse a single hop.

Besides, providing low latency communication the network should also provide high throughput. *Throughput* is the maximum traffic accepted by the network. High throughput network indicates that the network can accept maximum traffic before all the packets start experiencing high latencies.

The cost associated with NoCs is calculated in terms of area and power. Large scale multi-core processors operate under tight power budget. Hence, NoCs also work under strict power envelopes.

3. RELATED WORK

Recently significant research has been carried out to apply power-gating techniques to NoCs to minimize on-chip power consumption. Annavaram et al. [16] identified that the power gating mechanisms applied are prone to performance degradation beyond acceptable limits and might result in negative power savings. Thus, power gating techniques should be guarded against such holes. Kim et al. [17] and Bokhari et al. [18] proposed to save link and NoC dynamic power using voltage/frequency scaling.

Buffers in routers consume 55% of the static power while other router components consume 45% of the static power (17% of the total power) [8]. These observations imply that besides buffers, other router components also contribute significantly to the on chip static power consumption. Soteriou et al. [19], Matsutani et al. [20], Kim et al. [21] and Parikh et al. [10] proposed fine grained power-gating of components inside the NoC router. But such approaches require significant additional power-gating circuitry. These approaches work well to reduce the static power consumption, however, they only power-gate certain components (e.g. input buffer) of a router.

In [22], Chen et al. introduced a performance-aware, non-blocking power-gating scheme that wakes up powered-off routers along the path of a packet in advance, thereby preventing the packet from suffering router wakeup latency. Catnap [23] proposed a mechanism where a light-weight subnetwork can be power-gated based on the priority and predicted traffic load. This work is orthogonal to FLOV, since FLOV can be applied on top of the powered-on sub-networks to achieve even more power savings.

Chen et al. [8] proposed a node-router decoupling (NoRD) approach to leverage the independence of power-gating a core and its attached router. They provide a decoupling bypass route that connects the ejection and injection channels to form a bypass link to the router. The decoupling bypass links ensure network connectivity even for the extreme cases of all routers being turned off by using an escape ring network. However, a bypass ring is not scalable to large network sizes. Another issue with NoRD is that a bypass can be constructed in a $(k \times k)$ mesh, if and only if k is

even.

Zhan et al. [24] propose a mechanism that can activate powered down cores for performance gains while considering thermal aware floor planning and to this order they also explore topological/routing support.

3.1 Router Parking

Samih et al. [13] proposed Router Parking (RP) to power-gate as many routers as possible when their attached cores are sleeping. A centralized Fabric Manager is responsible for configuring, monitoring and re-configuring the interconnect to maintain network connectivity. FM could either be added as a functionality to either the firmware or one of the on-chip memory controllers. RP reconfigures the network after periodic time intervals called epochs to maintain connectivity. After every epoch, FM gathers information regarding any change to the state of nodes attached to the active routers. Based on the information obtained, FM decides to park (or power-gate) routers dynamically either using RP Aggressive or RP Conservative algorithms to maintain a balanced trade-off between power saving and performance. FM then updates the network configuration accordingly. However, this scheme is heavily dependent on the centralized Fabric Manager (FM) and typically takes a long time to reconfigure the network that may suspend new injections into the network during this phase.

3.2 FLY-OVER

Unlike previous studies, Boyapati et al. [14] propose FLOV mechanism, a distributed power gating technique to minimize static power leakage in routers. This eliminates the dependency on a centralized control to power gate routers. In a distributed power gating technique, power gating decision is made locally at the router through interaction with the neighboring routers. The key challenge to take care of, while implementing a distributed power gating mechanism is to maintain a consistent network state. To address this problem, FLOV has proposed a slightly modified router architecture, associated handshake protocols and routing algorithm to maintain network connectivity in the distributed scheme of power-gating routers.

3.2.1 FLOV Router Architecture

The FLOV router architecture is provided with flow links and a latch in all the four directions besides the baseline 3-stage virtual-channel router [15]. FLOV links in a router act as a simple connector between the upstream and downstream routers, thus making them logical neighbors for credit-based flow control. Multiplexers/Demultiplexers are provided to select the baseline router in the powered-on state and the latches in the power-gated state. In power-gated state, routers do not execute route computation or arbitration. Incoming flits are stored in the latch and forwarded to the downstream router in the next cycle based on the precomputed route by the upstream powered-on router. Handshake mechanism is used to communicate the power state transition to the neighboring routers. FLOV routers are also provided with Power State Registers (PSRs) and Handshake Control Logic (HSC) block to store the power states of the neighboring routers and implement the handshake protocols respectively.

Boyapati et al. [14] have proposed an elaborate power state transition scheme for power-gating the routers. They have used out-of-band signals to communicate with the neighboring routers such that the neighboring routers could update their PSRs based on the power state transition of the router being power-gated. The handshake between the neighboring routers is meticulously designed to maintain network connectivity. An *Active* or powered-on router enters the *Draining* state and then consecutively to the *Sleep* state as described below. A router in the *Sleep* state first moves to the *Wakeup* state and later to the *Active* state during the power-on phase.

On receiving the power-gating signal from the core, its attached router, call it S, waits for a few cycles to complete the transmission of packets coming into the router or going to the core. It then sends out a *drain* signal to its neighbors indicating it is in the *Draining* state. It also starts draining out its input buffers simultaneously. If two neighboring routers contend to proceed to *Draining* state, the router with the lower ID wins. This implies, the router with the higher ID moves back to the *Active* state. A transition from *Draining* state back to the *Active* state might also occur if the time spent by a router in the *Draining* state exceeds a predetermined *drain_threshold*.

The neighboring routers on receiving the *drain_signal* from the source router S, finish any

intermittent transfers destined to router S. Neighboring routers can not initiate new packet transfers after receiving the *drain* signal. On completion of transfer, neighboring routers send out a *drain_done* signal to router S. Router S, on receiving the *drain_done* signals from all the neighboring routers and emptying its input buffers, moves to *Sleep* state. On entering the *Sleep* state, the router now sends out a *Sleep* signal to its neighbors communicating that it has entered the *Sleep* state and so the neighboring routers can initiate new packet transfers. In the *Sleep* state, the router turns off the baseline router pipeline and transmits the incoming packets to the downstream routers through the latch on the FLOV link. In this state, router S, relays credits between the powered-on routers.

A router in the *Sleep* state transits to *Wakeup* state when its core is powered on or if any of its neighbors has a packet destined to it. Similar to the *Draining* state, router in *Wakeup* state sends out a *Wakeup* signal to its neighboring routers. The neighboring routers complete the ongoing packet transmissions while the waking up routers drains out the packets in the FLOV latch. Router in the *Wakeup* state also relays credits from the downstream to the upstream router. On receiving the *drain_done* signal from the downstream routers and emptying the latches, the router gets back to the *Active* state resuming the baseline router operations. It sends out an *Active* signal to the neighboring routers indicating its transition back to the *Active* state. Following this, the downstream routers update their credit counters to full availability.

3.2.2 FLOV Handshake Protocols

FLOV aims to achieve power efficiency with performance-aware considerations. FLOV has proposed two handshake protocols: restricted FLOV (rFLOV) and generalized FLOV (gFLOV). rFLOV imposes a constraint on the network that no two consecutive routers in a row/column can be powered down. Under this restriction, it can thus happen that router attached to a power-gated core could not be power-gated. As such, the power saving under rFLOV is limited. gFLOV on the other hand, lifts this restriction on the network for higher power savings. But this results in an added complexity of handshaking between series of power-gated routers to maintain consistent PSRs and credit control information. rFLOV has a relatively simpler implementation as the

handshake happens between adjacent physical neighbors. gFLOV also imposes additional protocol level restrictions to ease the handshake implementation [14].

3.2.3 Dynamic Routing Algorithm

Boyapati et al. [14] have also proposed a dynamic routing algorithm to route the packets through the irregular network 2D mesh network. To maintain connectivity in this irregular network, one of the columns/rows of the network is maintained in Always-ON (AON) state. Route is computed based on this algorithm only at the powered-ON routers while the power gated routers simply forward the packets straight across the router. Every router divides the network into eight partitions starting from the top right corner and moving in the anti-clockwise direction and numbered from 0 to 7 respectively. Routing algorithm is based on two variables: the partition in which the packet is destined to and the power state of the neighboring routers.

For the packets destined in partitions 1, 3, 5 and 7; the router routes them directly to Y+, X-, Y- and X+ directions respectively. FLOV links would ensure connectivity to the destination routers despite the power-gated routers in the path. Packets destined in partitions 0, 2, 4, and 6 would involve a turn to the destination. Packet delivery to the destination can not be guaranteed as the power-gated routers do not involve in router computation. Based on YX routing scheme, the packet is sent to the Y direction but if the router in the Y direction is power-gated, the packet is forwarded to X direction. But if the routers in both Y and X directions are power-gated, the packet is routed towards the AON routers where the packet could make a turn towards the destination router. Routing back to the same row again at the AON routers is prohibited to avoid livelock situations.

However, this adaptive routing algorithm is not completely deadlock free. Duato's algorithm and timeout mechanism are used for deadlock recovery. As per Duato's algorithm [25], one VC in every router is reserved as Escape VC for deadlock recovery. Escape VCs together form the Escape sub-network. Deterministic routing is followed in the escape network that ensures packet delivery to the destination router. In the escape sub-network, packets with destinations in partitions 1, 3, 5 and 7 are directed toward Y+, X-, Y- and X+ directions respectively, while the packets with

destinations in partitions 0, 2, 4 and 6 are directed towards the AON column. Packets stuck in regular VC sub-network for a longer duration than the pre-determined threshold are directed to the escape sub network to guarantee the packet delivery to the destination. Also, turns from south towards east, from east towards north, from east towards south and from north towards east are not allowed to ensure deadlock freedom.

4. BEST-EFFORT MINIMAL ROUTING ALGORITHM FOR FLY-OVER

4.1 Motivation

FLOV routing algorithm works well with low to medium fraction of routers power-gated. As the power-gated routers continue to increase, more packets will be directed to escape sub-network. This might cause low regular VC utilization and high congestion in escape sub-network especially the *AON* column. FLOV routing thus incurs unnecessary packet detours and as such the packets would not be routed through shortest path in some cases.

Packets with destinations located in partitions 0, 2, 4, and 6, would involve a turn to the destination. But, with the increase in fraction of power gated routers, it is highly likely that the immediate physical neighbors would be power-gated and the packet would be routed to *AON* column to make a turn towards the destination as per the existing FLOV routing algorithm. Intuitively, the packet could also make a turn at the nearest active router on the path rather than being directed to the escape sub-network to make a turn. This would evenly balance the network traffic and relieve the congestion in the *AON* column. Moreover, the routing algorithm could also exploit the power state information of the nearest logical neighbors, stored in PSRs, to make an informed routing decision to achieve the shortest path to the destination. In case the logical neighbors are power-gated, then the packet could be directed to the destination through the escape sub-network. This idea builds the motivation to explore an improved version of the existing routing algorithm, *Best-Effort Minimal Routing Algorithm* for FLOV to achieve minimal path routing.

4.2 Key Idea

Figure 4.1(a) and (b) show the sub-optimal and optimal routing examples, respectively. In (a), a packet is sent from Router 9 to Router 0 using FLOV routing algorithm. Destination is located in partition 2 and it involves a turn to the destination. Since both physical neighbors of source router are power-gated, the packet is directed to East for escape network all the way to *AON* column where it makes turns to reach the destination, leading to 7 hops in total. Note that there exists a

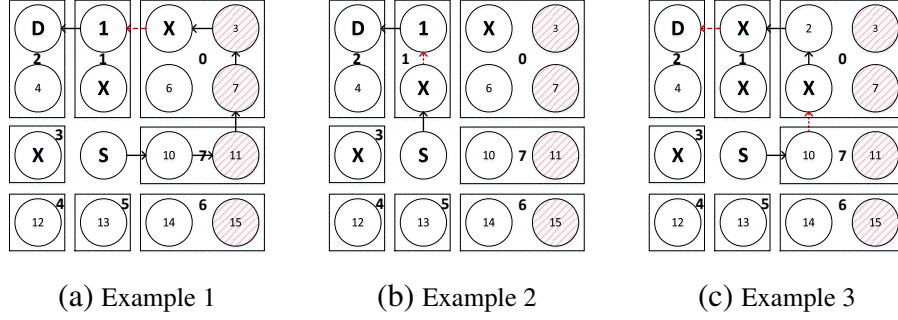


Figure 4.1: Routing Algorithm Examples: X indicates a power-gated router, Example 1 uses FLOW Routing, Example 2 and 3 use *Best-Effort Minimal Routing*.

viable shortest path from Router 9 to Router 0 by flying over power-gated Router 5 to reach Router 1 and make a turn there and travel only 3 hops as shown in Figure 4.1 (b). However, FLOW routing cannot take it due to the ignorance of the relative position between downstream *Active* router and destination.

To tackle the aforementioned problem, this work proposes an improved version of the existing routing algorithm, *Best-Effort Minimal Routing Algorithm* by exploiting the information of destination's position relative to the nearest active downstream router or the downstream logical neighbor's position. FLOW router microarchitecture holds the power state of the logical neighbors in a set of PSRs. The proposed algorithm also uses a partitioned-based dynamic routing algorithm based on YX routing for packets in regular VCs, similar to FLOW routing algorithm. The routing decision is made based on two variables, the partition which the destination falls into and power state of the logical neighbor.

For packets with destinations in partitions 1, 3, 5, and 7, the router will send them directly to North(Y+), West(X-), South(Y-), and East(X+) downstream routers, respectively similar to the FLOW routing algorithm. For packets with destinations in partitions 0, 2, 4, and 6, the route will include a turn towards the destination. In the proposed dynamic routing algorithm, if the logical neighbor in the Y direction is powered-on, the packet will be sent to this router using YX routing. If the logical neighbor in the Y direction is power gated, the router will check for a powered on logical neighbor in the X direction, and if it exists, the source router will send the packet to it. In

case the logical neighbors in both X and Y directions (besides the AON routers) are power gated, the packet is routed towards the AON column.

Note that during handshaking, router switching to *Sleep* state need to send its corresponding logical downstream neighbor's power state in each direction to its upstream router. Additionally, the handshaking was augmented to transfer the logical downstream neighbor's power state also. Therefore, downstream routers' relative positions to the destination will be known and better routing decisions can be made. The proposed routing algorithm is described in Algorithm 1. With this modification, it can also relax the burden of escape network, especially the *AON* column. For FLOV escape routing, a packet is forwarded to *AON* column to make turns which can lead to congestion. In contrast, when using *Best-Effort Minimal Routing* for FLOV, as shown in Figure 4.1(c), the packet can make a turn at Router 10 and fly over Router 6 to reach Router 2, finally turn to West to the destination. This routing path mitigates the pressure of *AON* column and is the minimal path in the irregular power-gated network.

Algorithm 1: Best-Effort Minimal FLOV Routing Algorithm.

Input: *cur, dest, in_port, in_vc, inqueue_time*

Output: *out_port, vc_set*

```
1 bool escape = false;
2 if in_vc == escape_vc || inqueue_time > threshold then
3     escape = true;
4     if dest in partition 1 then
5         | out_port = North;
6     else if dest in partition 5 then
7         | out_port = South;
8     else
9         | out_port = GetMinimalY(cur, dest);
10        | if logical_neighbor[out_port] not on minimal path then
11        |     | out_port = East;
12 else
13     if dest in partition 1 then
14         | out_port = North;
15     else if dest in partition 5 then
16         | out_port = South;
17     else if dest in partition 3 then
18         | out_port = West;
19     else if dest in partition 7 then
20         | out_port = East;
21     else
22         | out_port = GetMinimalY(cur, dest);
23         | if logical_neighbor[out_port] not on minimal path then
24         |     | out_port = GetMinimalX(cur, dest);
25         |     | if logical_neighbor[out_port] not on minimal path then
26         |     |     | out_port = East;
27         |     | if out_port == in_port then
28         |     |     | escape = true;
29         |     |     | out_port = East;
30 if escape == false then
31     | vc_set = regular_vc;
32 else
33     | vc_set = escape_vc;
34 return (out_port, vc_set);
```

5. EXPERIMENTAL EVALUATION

Modeling architectural hypothesis in software based simulators is extensively used in the world of computer architecture research. It helps in evaluating a design before building expensive hardware systems. Simulators also enable in obtaining detailed performance metrics and quicker debugging of design failures. Before proceeding to evaluate a new design idea, a baseline configuration is decided and suitable workloads to test the system are also selected. Performance metrics obtained on the baseline model serve as reference for comparison against metrics for the proposed design. Simulators cater to different level of implementation details and simulator should be chosen based on the scope of evaluation.

5.1 Simulator Configuration

FLOV power gating mechanism requires microarchitectural modifications to the router design. Thus, NoC module is evaluated at microarchitectural level in a cycle-accurate Booksim [26] simulator. DSENT [9] is used to estimate static and dynamic power consumptions of the interconnect components with a switching activity of 50% in 32nm technology.

Booksim is a cycle-accurate interconnection network simulator. A cycle-accurate simulator is a software based program that simulates a microarchitecture on a cycle-by-cycle basis. Cycle accurate simulators depict the system functionality accurately but are time-consuming as they update the state of the system at every cycle much like the hardware implementation.

Booksim is modeled to implement a FLOV router microarchitecture as described in section 2.1.2. rFLOV+ and gFLOV+ mechanisms are modeled in Booksim as a part of this work. Booksim is configured to select one these models based on a user defined flag given at run-time. These schemes are evaluated for different traffic pattern and traffic workload. Performance and power savings for these models are evaluated and analyzed for two types of synthetic workloads; Uniform Random and Tornado traffic for different injection rates. For Uniform Random traffic, each source sends an equal amount of traffic to every destination router. For Tornado traffic, traffic is only directed

to routers in the same row/column. Injection rate is the rate at which packets are injected into the simulator and the injection rate is specified in packets per flit cycle. For example, an injection rate = 0.25 means that each source injects a new packet in one out of every four simulator cycles.

At the start of the simulation, all the routers are assumed to be powered-on. For synthetic traffic, power-gated routers are selected at random based on the fraction of power-gating, that shall be powered down during the course of the simulation. Based on the power-gating mechanism, these selected routers are power-gated in the increasing order of their IDs. Also, the first 10,000 cycles were used to warm up the simulation and 100,000 cycles were run in total.

Table 5.1: Simulation Testbed Parameters

Network Topology	8×8 Mesh
Input Buffer Depth	6 flits
Router	3-stage (3 cycles) router
Virtual Channel	3 regular VCs and 1 escape VC per vent, 3 vnets
Packet Size	4 flits/packet for synthetic workload
Memory Hierarchy	32KB L1 I/D \$, 8MB L2 \$ MESI, 4 MCs at 4 corners
Technology	32nm
Clock Frequency	2GHz
Link	1mm, 1 cycle, 16B width
Power-Gating Parameters	Power-Gating overhead = 17. 7pJ wakeup latency = 10 cycles
Baseline Routing	YX Routing

Table 5.1 summarizes the simulation configuration parameters. A 2GHz clock frequency is assumed for the routers and links.

This work compares and analyzes the results from the following designs: (a) Baseline: Baseline design with no power-gating; (b) rFLOV: FLOV power-gating mechanism with restricted handshake protocol and FLOV routing algorithm; (c) gFLOV: FLOV power-gating mechanism with generalized handshake protocol and FLOV routing algorithm; (d) rFLOV+: FLOV power-gating mechanism with restricted handshake protocol and *Best-Effort Minimal Routing Algorithm*; and (e)

gFLOV+: FLOV power-gating mechanism with generalized handshake protocol and *Best-Effort Minimal Routing Algorithm*. In the discussion forward, *rFLOV schemes* refer to both rFLOV and rFLOV+. Similarly, *gFLOV schemes* refer to both gFLOV and gFLOV+.

6. RESULTS AND ANALYSIS

This chapter presents performance analysis for all the schemes under evaluation for synthetic workloads. Performance and cost parameters discussed in Section 2.2 are analysed for each of the schemes with an objective of maximum performance gain with minimum cost overhead.

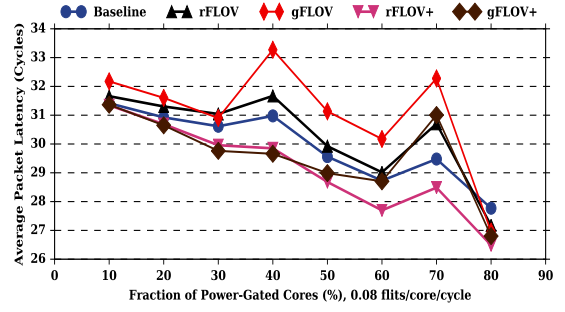
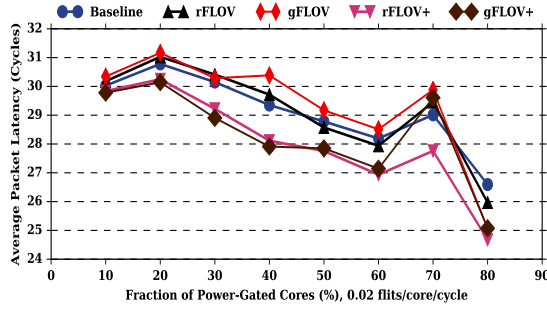
6.1 Latency Analysis

NoCs offer low latency communication solution for large scale CMPs. Thus, achieving minimum average packet latency would be criteria for the analysis presented below.

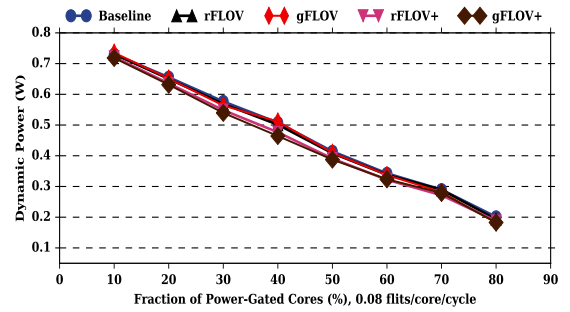
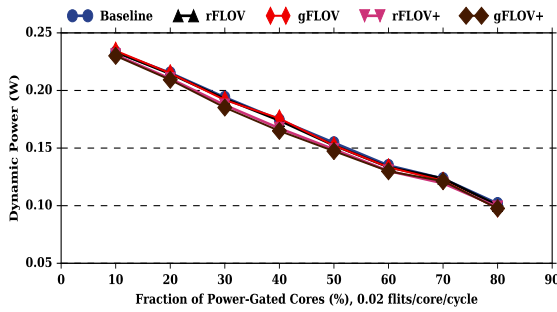
Figure 6.1 summarizes the simulation results using Uniform Random traffic. Similarly, Figure 6.3 shows the results for Tornado traffic. In the figures 6.1, 6.3; the first column depicts results for the injection rate of 0.02 flits/cycle/core while the second column is for the injection rate of 0.08 flits/cycle/core. Each row shows average latency, dynamic, and total power consumptions for a given injection rate, respectively with the fraction of power gated routers ranging from 10% to 80%. Figures 6.2 and 6.4 break down average packet latencies of the different mechanisms into router latency (number of hops \times router pipeline latency), link latency (total link traversals), serialization latency (number of flits per packet), contention latency (latency due to arbitration for switches or lack of credit i.e. waiting for credit from logical neighbors), and FLOV latency (number of FLOV routers/links traversed).

As the number of power-gated cores increases, rFLOV+, like rFLOV, also power-gates as many routers as possible under the aforementioned restrictions, and gFLOV+, like gFLOV, power-gates all the routers attached to the power-gated cores. Thus, with the increasing fraction of power-gated routers, the average packet latency for FLOV mechanisms is expected to decrease due to the utilization of fast FLOV links to fly over the power-gated routers. FLOV link utilization avoids 3-cycle baseline router per-hop latency, since the flit is temporarily held in the FLOV latch for once cycle, thereby reducing the packet latency.

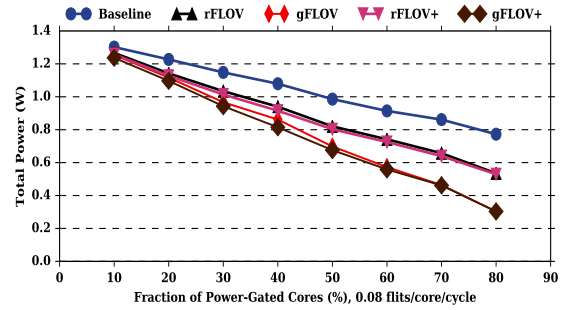
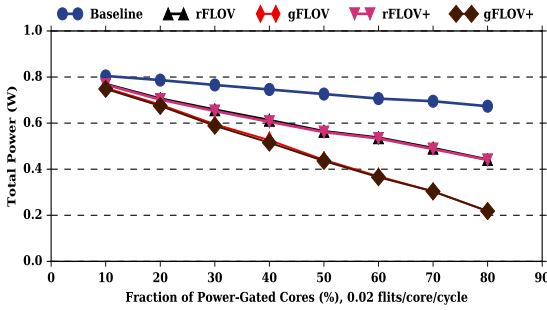
FLOV+ achieves lower average packet latency than FLOV routing as can be seen in Fig-



(a) Average Latency



(b) Dynamic Power Consumption



(c) Total Power Consumption

Figure 6.1: Average Latency and Power Comparison for Injection Rates of 0.02(left) and 0.08(right) flits/node/cycle with Uniform Random Traffic.

ure 6.1(a). This is mainly due to the fact that FLOV+ has higher chance to route packets through shortest path instead of sending to *AON* column to make turns. Therefore, it can reduce the number of hops traversed with respect to FLOV routing. Such benefit is confirmed by Figure 6.2

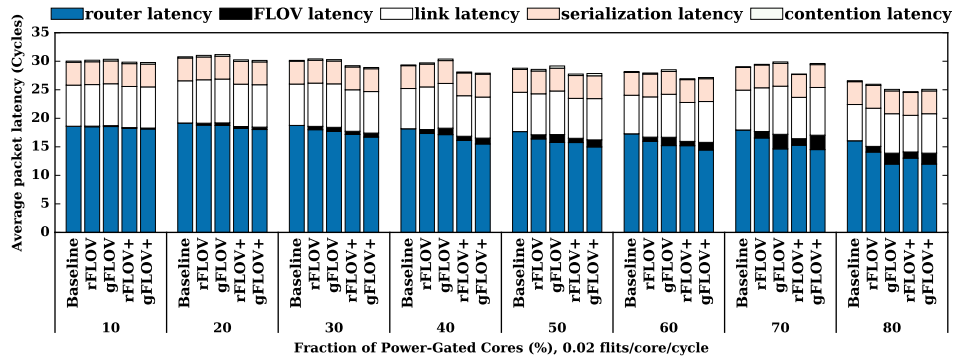
and Figure 6.4, where rFLOV+ and gFLOV+ have lower router latency than rFLOV and gFLOV respectively across the all fractions of power-gated routers.

As per Figure 6.1, at low fraction of power-gated routers, average packet latency for rFLOV and gFLOV is quite similar due to nearly equal number of power gated routers. gFLOV+ slightly performs better than rFLOV+ here due to availability of higher number of active routers to make a turn resulting in a reduced router latency and also utilization of FLOV links. As we move towards moderate fraction of power-gated routers, the network has a fair mix of active and power-gated routers. Figure 6.1 shows that average packet latency suddenly surges for both rFLOV and gFLOV schemes at 40% power gated routers for injection rate of 0.08. This can be attributed to a) packet detours in generalized FLOV power-gating mechanism indicated by the higher router and FLOV link utilization compared to that at 30% and b) increase in the contention latency as seen in Figure 6.2. However, the improved routing algorithms prevent these packet detours and facilitate minimal path routing as the packet latency continues to decrease for rFLOV+ and gFLOV+ as seen in Figure 6.1.

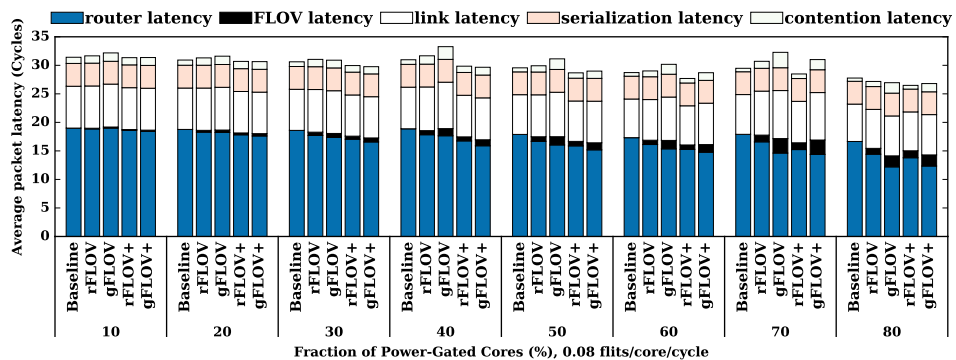
It is also interesting to note that, with 70% of the routers power-gated in the network, gFLOV+ is highly impacted as its packet latency overshoots the baseline configuration. This counterintuitive behavior can be accounted to packet detours due to a sparse network with limited active routers. As a result, gFLOV+ might have routed most of the traffic to the AON column like gFLOV as it probably could not have located active routers in the path to make a suitable turn to the destination. Figure 6.2 also indicates an increase in contention latency at 70% power-gating. This could be due to the credit waiting overhead as logical neighbors are located far from one another in a sparse network at high fractions of power-gating. Also, at higher fraction of power-gated routers, rFLOV+ performs the best as it could take advantage of the higher number of active routers in the network compared to gFLOV+ due to the aforementioned restriction on rFLOV+.

Interestingly, the improved routing algorithm also helps FLOV achieve better latency performance than Baseline owing to the fast FLOV links and minimal path routing. FLOV links incur one-cycle delay compared to a three-cycle delay through the router pipeline. With FLOV links on

the minimal path, the router pipeline stages can be avoided for faster packet delivery which can not be done in Baseline. One exception to this trend is observed at 70% of power-gated routers as discussed above. Average packet latency follows similar trend across different injection rates as evident from Figure 6.1.



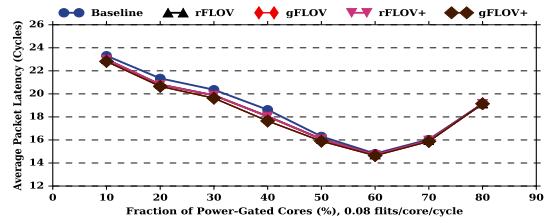
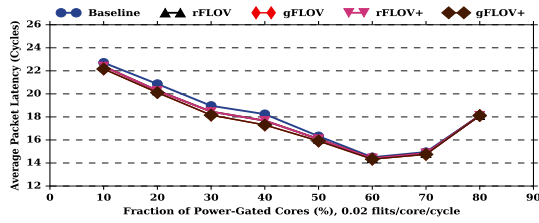
(a) Uniform Random Traffic Pattern (0.02 flits/cycle/core)



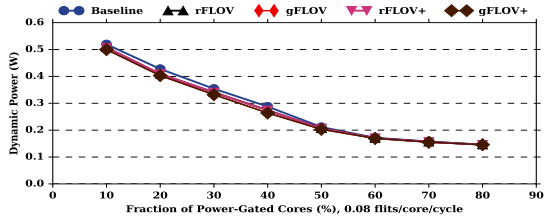
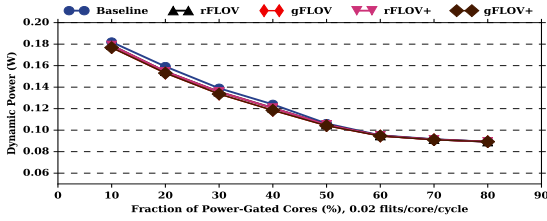
(b) Uniform Random Traffic Pattern (0.08 flits/cycle/core)

Figure 6.2: Packet Latency Breakdown For Uniform Random Traffic with Injection Rate of 0.02 (a) and 0.08 (b) flits/cycle/core.

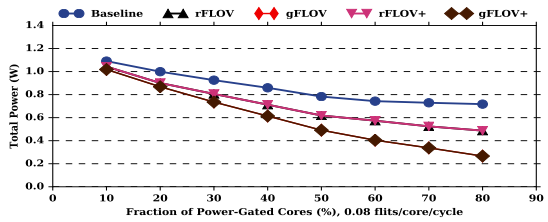
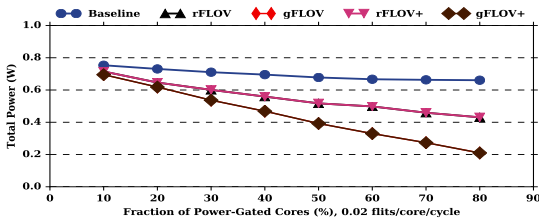
In Figure 6.3 (a), rFLOV+ and gFLOV+ perform quite similar to rFLOV and gFLOV respectively with Tornado traffic. This is because in Tornado, a significant portion of the traffic injected from each router is destined to a router in the same row/column i.e. the destination router is located in partitions 1, 3, 5 and 7. rFLOV+ and gFLOV+ are beneficial for packets destined in partitions 0, 2, 4, 6 as they include a turn to the destination. Thus, rFLOV+ and gFLOV+ do not provide sig-



(a) Average Packet Latency



(b) Dynamic Power Consumption

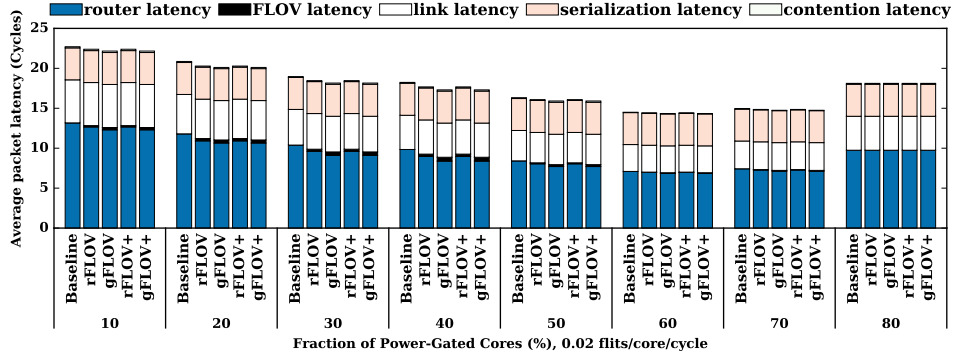


(c) Total Power Consumption

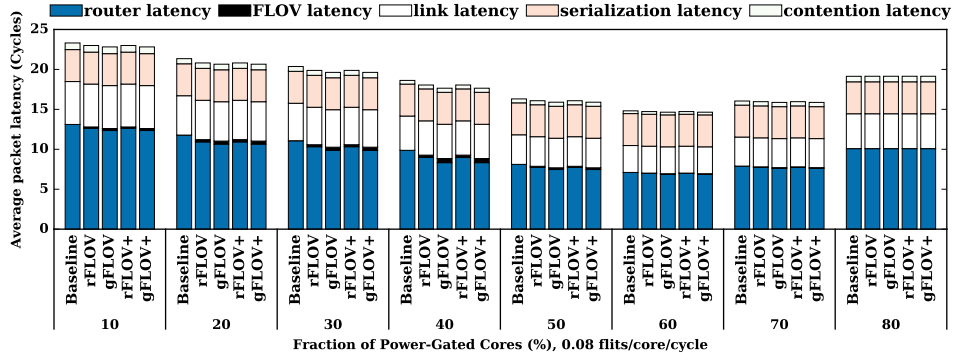
Figure 6.3: Average Latency and Power Comparison for Injection Rates of 0.02(left) and 0.08(right) flits/node/cycle with Tornado Traffic.

nificant advantage over rFLOV and gFLOV respectively for tornado traffic. Moreover, gFLOV(+) perform slightly better compared to rFLOV(+) due to higher FLOV link utilization owing to higher number of power gated routers in gFLOV(+).

In Figure 6.2(a), under Uniform Random traffic, the FLOV latency increases as more cores are power-gated for the FLOV mechanism, which shows the increased FLOV link utilization. For Tornado traffic in Figure 6.4(c), the communication occurs between two power-on nodes in the same row/column, and the routers in the rightmost column are always active. Therefore, less number of FLOV links are used, which leads to reduced FLOV latency. This analysis indicates that the performance of FLOV+ is highly sensitive to the traffic patterns and the configuration of



(a) Tornado Traffic Pattern (0.02 flits/cycle/core)



(b) Tornado Traffic Pattern (0.08 flits/cycle/core)

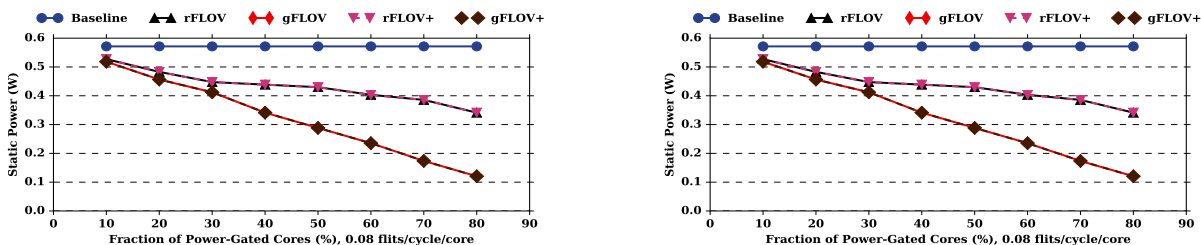
Figure 6.4: Packet Latency Breakdown For Tornado Traffic with Injection Rate of 0.02 (a) and 0.08 (b) flits/cycle/core.

power-gated routers in the network.

In Figure 6.2 (b) and 6.4 (b), both rFLOV and gFLOV have relatively higher contention latency compared to rFLOV+ and gFLOV+ respectively, at high fractions of power-gated cores. One reason is that packets have higher probability of being routed to the AON router column for guaranteed paths to the destinations, which may create congestion in the AON router column. Also, when packets are routed through consecutive FLOV links in a row/column, packet transmission may be delayed due to the round-trip latency of credit information. FLOV+ reduces the contention latency, evenly distributing the traffic and avoiding congestion in the AON column as can be seen in Figure 6.2.

Best-Effort Minimal Routing Algorithm performs better than existing FLOV routing algorithm

across different traffic and injection rates. This is because, even though FLOV mechanism with FLOV routing tries to route a packet through a minimal path using FLOV links, it does not fully exploit the nearest available active router to make a turn to the destination to avoid packet detours and network congestion. FLOV+ takes advantage of both the FLOV links and the power state of the logical neighbor to achieve minimal routing. This can be observed clearly in Figure 6.2(a) and (b), where the router latency for FLOV+ schemes is the smallest among all the schemes evaluated in this work resulting in minimum average packet latency.



(a) Uniform Random Traffic Pattern (0.02 flits/cycle/core) (b) Uniform Random Traffic Pattern (0.08 flits/cycle/core)

Figure 6.5: Static Power Comparison of FLOV with RP and Baseline.

6.2 Power Consumption

Figures 6.1 (b, c), 6.3 (b, c) show dynamic and total power consumptions of FLOV+ schemes in comparison FLOV with the existing FLOV routing algorithm and Baseline for multiple injection rates. In Figures 6.1 (b) and 6.3 (b), for multiple injection rates the dynamic power consumptions of rFLOV+ and gFLOV+ are lower than rFLOV and gFLOV respectively, since router latency of FLOV+ is lower than the existing FLOV schemes, and thus, lower dynamic power is consumed. Both rFLOV+ and gFLOV+ have better power savings than baseline, as they take advantage of both the FLOV links and the improved minimal routing algorithm. Dynamic power saving is impacted by the packet latency. As the reduction in packet latency for Tornado traffic is less compared to Uniform Random traffic, the dynamic power saving for Tornado traffic is lesser compared to

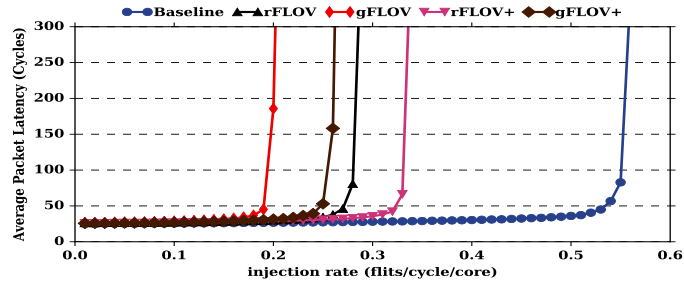
Uniform Random traffic distribution as confirmed by Figures 6.1 (b) and 6.3 (b)

Figure 6.5 shows static power consumption comparison, which is injection rate and workload independent for FLOV based power-gating mechanisms. gFLOV(+) power-gates the routers attached to power-gated cores, while rFLOV(+) power-gates a limited number of routers to preserve the restriction. As such, gFLOV has better power savings than rFLOV. As static power consumption does not depend on the routing algorithm, FLOV+ has the exactly the same power savings as FLOV.

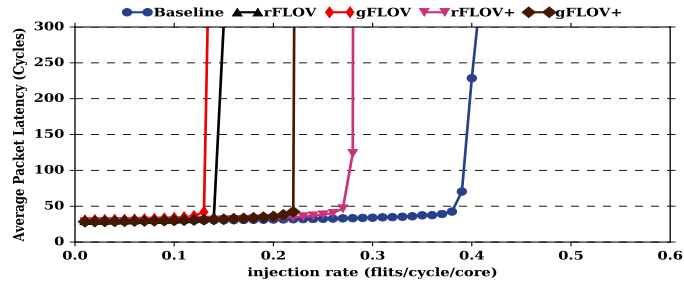
Figures 6.1 (c) and 6.3 (c) show total power consumptions of rFLOV+ and gFLOV+ compared with rFLOV and gFLOV. It is clear that gFLOV+ unanimously has lower power consumption, since the dynamic and static power consumptions in gFLOV+ are lower than the rest. The static power saving rFLOV+ and gFLOV+ is same as rFLOV and gFLOV respectively as it solely depends on the number of power-gated routers. The power saving in dynamic power consumption FLOV+, thus, reflects as the power savings in the total power consumption.

6.3 Throughput and Scalability Analysis

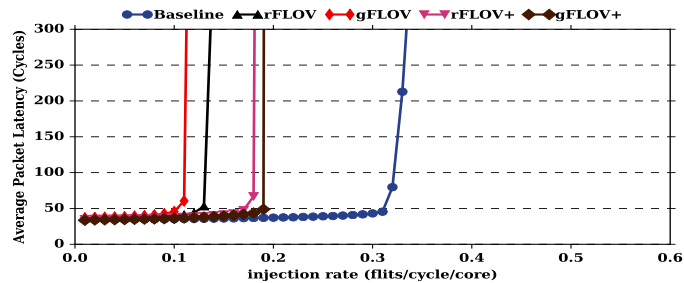
Figure 6.6 shows the load-latency curves under Uniform Random traffic with 50% cores power-gated. As shown in Figure 6.6 (b) for a 8x8 mesh network, the throughput of gFLOV saturated earliest, at about 0.12 flits/cycle/core, it is because these schemes power-gates most of the cores and cannot provide sufficient service rate to sustain more packets. Then, rFLOV saturates next as it could service more packets compared to gFLOV as it maintains higher number of active cores. With help of routing optimization, gFLOV+ and rFLOV+ have the highest throughput among all power-gating mechanisms as they evenly distribute traffic and aid in reducing the network congestion. Compared to gFLOV+, rFLOV+ postpones the saturation point later. For a power-gating mechanism, it can adapt to the traffic and adjust its power saving aggressiveness to transit from different power-gating mode in order to provide good performance. The behaviors shown in Figure 6.6 imply that gFLOV+ and rFLOV+ can provide guaranteed performance while power-gate as many cores as possible compared to other mechanisms, which is a performance-power win-win solution. Figure 6.6 also shows load-latency curves for different network dimensions.



(a) 6×6 Mesh



(b) 8×8 Mesh



(c) 10×10 Mesh

Figure 6.6: Load-Latency Curves for Uniform Random Traffic with 50% Cores Power-Gated for 6×6 (a), 8×8 (b) and 10×10 Mesh Network

As the network scales, the saturation points for gFLOV and rFLOV become close to each other. gFLOV+ and rFLOV+ scale well to saturate latest but their gap reduces as network scales. One interesting observation is that gFLOV+ surpasses the saturation point of rFLOV+ as in 10×10 scale shown in Figure 6.6 (c). This is because, as the network scales, the number of routers to be power-gated increases with same fraction of power-gated cores. With more number of power-gated routers, packets has higher chance to traverse through fast FLOV links in gFLOV+, which can serve the packets faster and effectively increase its saturation load.

7. CONCLUSION

Power gating is a promising technique that can be used to minimize static power consumption in NoCs. *Fly-Over* is a light weight distributed router power gating mechanism to reduce static power consumption in NoCs. FLOV power-gates routers attached to powered-down cores without global network information, but still ensures network connectivity through flow links. However, the existing FLOV routing algorithm introduces unnecessary detours and pressurizes the routers in *AON* column resulting in high packet latencies and network congestion. This work has proposed an improved routing algorithm, *Best-Effort Minimal Routing Algorithm* (FLOV+) that uses the power state information of the logical neighbors and routes the packets through shortest paths improving the utilization of the active routers in the network. This also aids in relieving the pressure on the *AON* column and evenly balances the network traffic minimizing the network congestion and packet latency.

FLOV+ was evaluated for both restricted FLOV and generalized FLOV handshake protocols for synthetic workloads. Experiments conducted as a part of this work show that FLOV+ reduces average packet latency upto 9.84% in an 8-dimensional mesh network. rFLOV+ and gFLOV+ were also found to achieve higher throughput compared rFLOV and gFLOV mechanisms respectively. FLOV+ has better throughput gains at higher dimension networks and thus scales well with the network size. Simulation results also show 50% and 40% improvement in the network throughput for rFLOV+ and gFLOV+ power gating mechanisms respectively. The reduction in average packet latency using FLOV+ has also resulted in about 4% reduction in the dynamic power consumption compared to the baseline configuration at no additional hardware cost over the baseline FLOV router.

REFERENCES

- [1] G. Moore, “Cramming More Components onto Integrated Circuits,” *Electronics*, vol. 38, no. 8, p. 56, 1965.
- [2] J. D. Owens, W. J. Dally, R. Ho, D. N. Jayasimha, S. W. Keckler, and L. S. Peh, “Research challenges for on-chip interconnection networks,” *IEEE Micro*, vol. 27, pp. 96–108, Sept 2007.
- [3] M. B. Taylor, J. Kim, J. Miller, D. Wentzlaff, F. Ghodrat, B. Greenwald, H. Hoffman, P. Johnson, J.-W. Lee, W. Lee, A. Ma, A. Saraf, M. Seneski, N. Shnidman, V. Strumpen, M. Frank, S. Amarasinghe, and A. Agarwal, “The Raw Microprocessor: A Computational Fabric for Software Circuits and General-Purpose Programs,” *IEEE Micro*, vol. 22, no. 2, pp. 25–35, 2002.
- [4] J. Howard, S. Dighe, S. R. Vangal, G. Ruhl, N. Borkar, S. Jain, V. Erraguntla, M. Konow, M. Riepen, M. Gries, G. Droege, T. Lund-Larsen, S. Steibl, S. Borkar, V. K. De, and R. Van Der Wijngaart, “A 48-Core IA-32 Processor in 45nm CMOS Using On-Die Message-Passing and DVFS for Performance and Power Scaling,” *IEEE Journal of Solid-State Circuits*, vol. 46, no. 1, pp. 173–183, 2011.
- [5] Y. Hoskote, S. Vangal, A. Singh, N. Borkar, and S. Borkar, “A 5-GHz Mesh Interconnect for a Teraflops Processor,” *IEEE Micro*, vol. 27, no. 5, pp. 51–61, 2007.
- [6] X. Chen and L.-S. Peh, “Leakage Power Modeling and Optimization in Interconnection Networks,” in *International Symposium on Low Power Electronics and Design (ISLPED)*, pp. 90–95, ACM, 2003.
- [7] A. Banerjee, R. Mullins, and S. Moore, “A Power and Energy Exploration of Network-on-Chip Architectures,” in *International Symposium on Networks-on-Chip (NoCS)*, pp. 163–172, IEEE Computer Society, 2007.

- [8] L. Chen and T. M. Pinkston, “NoRD: Node-Router Decoupling for Effective Power-Gating of On-Chip Routers,” in *International Symposium on Microarchitecture (MICRO)*, pp. 270–281, IEEE Computer Society, 2012.
- [9] C. Sun, C.-H. Chen, G. Kurian, L. Wei, J. Miller, A. Agarwal, L.-S. Peh, and V. Stojanovic, “DSENT – A Tool Connecting Emerging Photonics with Electronics for Opto-Electronic Networks-on-Chip Modeling,” in *International Symposium on Networks on Chip (NoCS)*, pp. 201–210, IEEE, 2012.
- [10] R. Parikh, R. Das, and V. Bertacco, “Power-Aware NoCs through Routing and Topology Reconfiguration,” in *Design Automation Conference (DAC)*, pp. 1–6, IEEE, 2014.
- [11] R. H. Dennard, F. H. Gaensslen, V. L. Rideout, E. Bassous, and A. R. LeBlanc, “Design of Ion-Implanted MOSFET’s with Very Small Physical Dimensions,” *IEEE Journal of Solid-State Circuits*, vol. 9, no. 5, pp. 256–268, 1974.
- [12] “Idc. future of virtualization.,” 2008.
- [13] A. Samih, R. Wang, A. Krishna, C. Maciocco, C. Tai, and Y. Solihin, “Energy-Efficient Interconnect via Router Parking,” in *International Symposium on High Performance Computer Architecture (HPCA)*, pp. 508–519, IEEE, 2013.
- [14] R. Boyapati, J. Huang, N. Wang, K. H. Kim, K. H. Yum, and E. J. Kim, “Fly-over: A light-weight distributed power-gating mechanism for energy-efficient networks-on-chip,” in *2017 IEEE International Parallel and Distributed Processing Symposium (IPDPS)*, pp. 708–717, May 2017.
- [15] L.-S. Peh and W. J. Dally, “A Delay Model and Speculative Architecture for Pipelined Routers,” in *International Symposium on High-Performance Computer Architecture (HPCA)*, pp. 255–266, IEEE, 2001.
- [16] M. Annavaram, “A Case for Guarded Power Gating for Multi-Core Processors,” in *International Symposium on High Performance Computer Architecture (HPCA)*, pp. 291–300, IEEE, 2011.

- [17] E. J. Kim, K. H. Yum, G. M. Link, N. Vijaykrishnan, M. Kandemir, M. J. Irwin, M. Yousif, and C. R. Das, “Energy Optimization Techniques in Cluster Interconnects,” in *International Symposium on Low Power Electronics and Design (ISLPED)*, pp. 459–464, ACM, 2003.
- [18] H. Bokhari, H. Javaid, M. Shafique, J. Henkel, and S. Parameswaran, “Malleable NoC: Dark Silicon Inspired Adaptable Network-on-Chip,” in *Proceedings of the 2015 Design, Automation & Test in Europe Conference & Exhibition (DATE)*, pp. 1245–1248, EDA Consortium, 2015.
- [19] V. Soteriou and L.-S. Peh, “Design-Space Exploration of Power-Aware On/Off Interconnection Networks,” in *International Conference on Computer Design (ICCD)*, pp. 510–517, IEEE, 2004.
- [20] H. Matsutani, M. Koibuchi, D. Ikebuchi, K. Usami, H. Nakamura, and H. Amano, “Ultra Fine-Grained Run-Time Power Gating of On-Chip Routers for CMPs,” in *International Symposium on Networks-on-Chip (NOCS)*, pp. 61–68, IEEE, 2010.
- [21] G. Kim, J. Kim, and S. Yoo, “Flexibuffer: Reducing Leakage Power in On-Chip Network Routers,” in *Design Automation Conference (DAC)*, pp. 936–941, IEEE, 2011.
- [22] L. Chen, D. Zhu, M. Pedram, and T. M. Pinkston, “Power Punch: Towards Non-Blocking Power-Gating of NoC Routers,” in *International Symposium on High Performance Computer Architecture (HPCA)*, pp. 1–12, IEEE, 2015.
- [23] R. Das, S. Narayanasamy, S. K. Satpathy, and R. G. Dreslinski, “Catnap: Energy Proportional Multiple Network-on-Chip,” in *ACM SIGARCH Computer Architecture News*, vol. 41, pp. 320–331, ACM, 2013.
- [24] J. Zhan, Y. Xie, and G. Sun, “NoC-Sprinting: Interconnect for Fine-Grained Sprinting in the Dark Silicon Era,” in *Proceedings of the 51st Annual Design Automation Conference (DAC)*, pp. 1–6, ACM, 2014.
- [25] J. Duato, “A New Theory of Deadlock-Free Adaptive Routing in Wormhole Networks,” *IEEE Transactions on Parallel and Distributed Systems*, vol. 4, no. 12, pp. 1320–1331, 1993.

- [26] N. Jiang, D. U. Becker, G. Michelogiannakis, J. Balfour, B. Towles, D. E. Shaw, J.-H. Kim, and W. J. Dally, “A Detailed and Flexible Cycle-Accurate Network-on-Chip Simulator,” in *International Symposium on Performance Analysis of Systems and Software (ISPASS)*, pp. 86–96, IEEE, 2013.