

DISSERTATION

RELIABLE, ENERGY-EFFICIENT, AND SECURE SILICON PHOTONIC
NETWORK-ON-CHIP DESIGN FOR MANYCORE ARCHITECTURES

Submitted by

Sai Vineel Reddy Chittamuru

Department of Electrical and Computer Engineering

In partial fulfillment of the requirements

For the Degree of Doctor of Philosophy

Colorado State University

Fort Collins, Colorado

Spring 2018

Doctoral Committee:

Advisor: Sudeep Pasricha

Anura Jayasumana

Sourajeet Roy

Yashwant K. Malaiya

Copyright by Sai Vineel Reddy Chittamuru 2018

All Rights Reserved

ABSTRACT

RELIABLE, ENERGY-EFFICIENT, AND SECURE SILICON PHOTONIC NETWORK-ON-CHIP DESIGN FOR MANYCORE ARCHITECTURES

Advances in technology scaling over the past several decades have enabled the integration of billions of transistors on a single die. Such a massive number of transistors has allowed multiple processing cores and significant memory to be integrated on a chip, to meet the rapidly growing performance demands of modern applications. These on-chip processing and memory components require an efficient mechanism to communicate with each other. Thus emerging manycore architectures with high core counts have adopted scalable packet switched electrical network-on-chip (ENoC) fabrics to support on-chip transfers. But with several hundreds to thousands of on-chip cores expected to become a reality in the near future, ENoCs are projected to suffer from cripplingly high power dissipation and limited performance. Recent developments in the area of silicon photonics have enabled the integration of on-chip photonic interconnects with CMOS circuits, enabling photonic networks-on-chip (PNoCs) that can offer ultra-high bandwidth, reduced power dissipation, and lower latency than ENoCs.

There are several challenges that hinder the commercial adoption of these PNoC architectures. Especially, the operation of silicon photonic components is very sensitive to thermal variations (TV) and process variations (PV) that frequently occur on a chip. These variations and their mitigation techniques create significant reliability issues and increase energy costs in PNoCs. Furthermore, photonic components are susceptible to intrinsic crosstalk noise and aging, which demands higher energy for reliable communication. Moreover, contention in photonic waveguides

as well as laser power distribution overheads also reduce performance and energy-efficiency. In addition, hardware trojans (HTs) in the electrical circuitry of photonic components lead to covert data snooping from shared photonic waveguides and introduces serious hardware security threats.

To address these challenges, in this dissertation we propose a cross-layer framework towards the design of reliable, secure, and energy-efficient PNoC architectures. We devise layer-specific solutions for PNoC design as part of our framework: (i) we propose device-level enhancements to adapt to TV, and to mitigate heterodyne crosstalk and intermodulation effect induced heterodyne crosstalk; we also analyze aging in photonic components and explore its impact on PNoCs; (ii) at the circuit-level we propose PV-aware homodyne and heterodyne crosstalk mitigation mechanisms, a PV-aware security enhancement mechanism, and TV- and PV-aware photonic component assignment mechanisms; (iii) at the architecture-level we propose new application specific and reconfigurable PNoC architectures to improve photonic channel utilization, a laser power management scheme across components of PNoC architectures, and a reservation-assisted security enhancement scheme to improve security in PNoC architectures; and (iv) at the system-level we propose TV and PV aware thread migration schemes and application scheduling schemes that exploit adaptive application degree of parallelism (DoP).

In addition to layer-specific enhancements, we also combine techniques across layers to create cross-layer optimization strategies to aggressively improve reliability and energy-efficiency in PNoC architectures. In our SPECTRA and LIBRA frameworks we combine system-level and circuit-level enhancements for TV management in PNoCs. In our ‘Island of Heater’ framework we combine system-level and device-level enhancements for TV management in PNoCs. We combine device-level and circuit-level enhancements for heterodyne crosstalk mitigation in our PICO and HYDRA frameworks. Our proposed BiGNoC architecture uses architectural-level

enhancements and system-level application scheduling to improve its performance and energy-efficiency. Lastly, in our SOTERIA framework we combine circuit-level and architecture-level enhancements to enable secure communication in DWDM-based PNoC architectures.

ACKNOWLEDGEMENTS

I would like to thank all the individuals whose encouragement and support have made the completion of this dissertation possible.

First and foremost, I would like to express my sincere gratitude to my advisor, Dr. Sudeep Pasricha, who has patiently guided me through the entire process of graduate study step by step. It is only with his encouragement and patience that I was able to survive the trial of graduate school while working on the new and exciting area of silicon photonic Network-on-Chip design for manycore architectures. In the last year of my bachelors program in Electrical Engineering, I made up my mind to seek an overseas study opportunity in another area to feed my curiosity about the interaction between computer hardware and software. Although the picture of snowcapped mountains on the Colorado State University ECE department website was impressive, it was Dr. Pasricha's description of research on multicore embedded systems that immediately caught my eye and enlightened me to the field I like. Since then I have never looked back as I was fortunate enough to join his research group and to receive his help that changed my life. In the first year, the coursework and research work suggested by Dr. Pasricha helped me prepare for the basic skills needed for research and reassured me that I had found my area of interest. After that, it was his vision and wisdom that stimulated me to look at research problems with more critical and creative thinking, which led to several publications in well-known conferences and journals. Over countless times, I was impressed by his thoroughness and attention to detail despite his tight schedule, from which I got to know his passion and enthusiasm for research. On the other hand, he is the type of advisor that is caring enough to suggest his graduate students to slow down, get some rest, and recharge whenever he senses high pressure on them. Dr. Pasricha can also give good life advice

when inquired, which helped me to overcome various difficulties and confusions in life and study during my graduate school years. I really appreciate all the help, guidance, and inspiration I received from Dr. Pasricha, who made it possible for me to survive the trial of graduate school with unforgettable memories and broadened horizons.

I would like to take this opportunity to thank the respected members of my PhD committee, Dr. Anura Jayasumana, Dr. Sourajeet Roy, and Dr. Yashwant K. Malaiya. Their feedback helped me to rediscover my research and refine my work from different perspectives. Furthermore, my special thanks to my research partner and dear friend Ishan Thakkar, whose collaboration helped me to broaden my research area by gaining valuable insights on nanophotonic devices. In addition, I would like to appreciate the research contributions of Dharanidhar Dang and his advisor Prof. Rabi Mahapatra of Texas A&M University. I am also thankful to my mates in Dr. Pasricha's EPIC lab for their collaboration during my Ph.D. study: Srinivas Desai, Daniel Dauwe, Yaswanth Raparti, Yi Xiang, Yong Zou, Nishit Kapadia, and Shirish Bahirat. Also this list cannot be complete without mentioning company and help from Vipin Kumar Kukkala, Varun Bhatt, Sai Kiran Koppu, Saideep Tiku, Vinay Ugave, Tejasi Pimpalkhute, Pramit Rajkrishna, and Shoumik Maiti.

Last but not least, I would like to thank my family, especially my father Sridhar Reddy Chittamuru, my mother Madhavi Chittamuru, and my wife Suveka Siddavarapu, for their support to pursue my Ph.D. I cannot wait to share more good news with them in the future as I continue with my work and study. Their kindness shaped my view of this world and made me the person I am.

TABLE OF CONTENTS

ABSTRACT.....	ii
ACKNOWLEDGEMENTS.....	v
TABLE OF CONTENTS.....	vii
LIST OF TABLES.....	xvi
LIST OF FIGURES.....	xviii
LIST OF ALGORITHMS.....	xxx
LIST OF RESEARCH PUBLICATIONS.....	xxxii
1. INTRODUCTION.....	1
1.1. MOTIVATION FOR CMP DESIGN.....	1
1.2. PHOTONIC INTERCONNECTS.....	3
1.2.1. PHOTONIC WAVEGUIDES.....	4
1.2.2. MICRORING RESONATORS.....	5
1.2.3. TRANS-IMPEDANCE AMPLIFIERS, COMBINERS, AND SPLITTERS.....	7
1.3. DESIGN CHALLENGES IN PNOCS.....	7
1.3.1. PERFORMANCE CHALLENGES.....	8
1.3.2. RELIABILITY CHALLENGES.....	8
1.3.3. POWER CHALLENGES.....	10
1.3.4. SECURITY CHALLENGES.....	11
1.4. DISSERTATION OUTLINE.....	11
2. SWIFTNOC: A RECONFIGURABLE SILICON-PHOTONIC NETWORK WITH MULTICAST ENABLED CHANNEL SHARING FOR MULTICORE ARCHITECTURES ..	17
2.1. MOTIVATION AND CONTRIBUTION.....	17

2.2. RELATED WORK	18
2.3. ULTRANOC AND SWIFTNOC : PHOTONIC ARCHITECTURE OVERVIEW	21
2.3.1. ULTRANOC ARCHITECTURE AND TERMINOLOGY	21
2.3.2. MWMMR CONCURRENT TOKEN STREAM ARBITRATION AND RECEIVER SELECTION IN ULTRANOC	26
2.3.3. IMPROVED MWMMR CONCURRENT TOKEN STREAM ARBITRATION IN SWIFTNOC.....	29
2.3.4. MULTICASTING OF MESSAGES IN SWIFTNOC	29
2.3.5. INTER-CLUSTER BANDWIDTH EXCHANGE IN SWIFTNOC.....	31
2.3.6. CLUSTER PRIORITY ADAPTATION WITH LSWC RECONFIGURATION	33
2.4. EXPERIMENTS	35
2.4.1. EXPERIMENTAL SETUP	35
2.4.2. EXPERIMENTAL RESULTS	40
2.4.2.1. SENSITIVITY ANALYSIS TO DETERMINE OPTIMAL RECONFIGURATION WINDOW SIZE	40
2.4.2.2. RESULTS OF 64-CORE SYSTEM FOR SYNTHETIC TRAFFIC.....	41
2.4.2.3. EXPERIMENTAL ANALYSIS WITH 64-CORE CMP	48
2.4.2.4. SCALABILITY ANALYSIS WITH 256-CORE CMP.....	52
2.4.2.5. SUMMARY OF RESULTS AND OBSERVATIONS	54
2.5. CONCLUSIONS.....	55
3. BIGNOC: ACCELERATING BIG DATA COMPUTING WITH APPLICATION- SPECIFIC PHOTONIC NETWORK-ON-CHIP ARCHITECTURES	56
3.1. BACKGROUND, MOTIVATION, AND CONTRIBUTION	56
3.2. RELATED WORK	59
3.3. MASTER-SERVANT CLUSTER ARCHITECTURE.....	61

3.3.1. MN-TO-SN COMMUNICATION IN MSNOC CLUSTER	64
3.3.2. SN-TO-MN COMMUNICATION IN MSNOC CLUSTER	69
3.3.3. SN-TO-SN COMMUNICATION IN MSNOC CLUSTER.....	71
3.4. MSNOC: SENSITIVITY ANALYSIS	71
3.5. BIGNOC ARCHITECTURE	73
3.5.1. HOMOGENEOUS BIGNOC ARCHITECTURE	73
3.5.2. HETEROGENEOUS BIGNOC ARCHITECTURE.....	76
3.5.3. APPLICATION SCHEDULING IN BIGNOC.....	78
3.6. EXPERIMENTS	79
3.6.1. EXPERIMENTAL SETUP	79
3.6.2. BIGNOC: SENSITIVITY ANALYSIS	82
3.6.3. EXPERIMENTAL RESULTS	84
3.7. CONCLUSIONS.....	89
4. CROSSTALK MITIGATION FOR HIGH-RADIX AND LOW-DIAMETER PHOTONIC NOC ARCHITECTURES	91
4.1. MOTIVATION AND CONTRIBUTION.....	91
4.2. RELATED WORK	92
4.3. ANALYTICAL MODELS FOR CROSSTALK ANALYSIS IN DWDM-BASED PNOC ARCHITECTURES	94
4.3.1. OVERVIEW OF MR OPERATION IN DWDM-BASED PNOCS	94
4.3.2. ANALYTICAL MODELS FOR CROSSTALK-NOISE AND SIGNAL-POWER... 95	
4.4. TECHNIQUES TO MITIGATE CROSSTALK NOISE	98
4.4.1. PCTM5B ENCODING TECHNIQUE.....	99
4.4.2. PCTM6B ENCODING TECHNIQUE.....	100
4.5. EVALUATION STUDIES	101

4.5.1. EVALUATION METHODOLOGY	101
4.5.2. EVALUATION RESULTS WITH CORONA ARCHITECTURE	104
4.5.3. EVALUATION RESULTS WITH FIREFLY ARCHITECTURE.....	105
4.5.4. SUMMARY OF RESULTS AND OBSERVATIONS.....	108
4.6. CONCLUSIONS.....	109
5. IMPROVING CROSSTALK RESILIENCE WITH WAVELENGTH SPACING IN PHOTONIC CROSSBAR-BASED NETWORK-ON-CHIP ARCHITECTURES.....	110
5.1. MOTIVATION AND CONTRIBUTION.....	110
5.2. RELATED WORK	111
5.3. WAVELENGTH SPACING (WSP) TECHNIQUE.....	112
5.3.1. ANALYTICAL MODEL FOR OSNR IN CORONA CROSSBAR-BASED PNOC	113
5.3.2. WAVELENGTH SPACING (WSP) TECHNIQUE.....	114
5.4. EXPERIMENTS	115
5.4.1. EXPERIMENTAL SETUP	115
5.4.2. EXPERIMENTAL RESULTS WITH CORONA AND FIREFLY PNOCS	117
5.4.3. SUMMARY OF RESULTS AND OBSERVATIONS.....	121
5.5. CONCLUSIONS.....	122
6. PICO: MITIGATING HETERODYNE CROSSTALK DUE TO PROCESS VARIATIONS AND INTERMODULATION EFFECTS IN PHOTONIC NOCS	123
6.1. MOTIVATION AND CONTRIBUTION.....	123
6.2. RELATED WORK	125
6.3. PV-AWARE CROSSTALK ANALYSIS.....	127
6.3.1. IMPACT OF LOCALIZED TRIMMING ON CROSSTALK.....	127
6.3.2. PV-AWARE CROSSTALK MODELS FOR CORONA PNOC.....	130

6.3.3. MODELING PV OF MR DEVICES IN CORONA PNOC.....	133
6.4. IM CROSSTALK ANALYSIS.....	133
6.5. IM-AWARE CROSSTALK MITIGATION.....	135
6.6. PV-AWARE CROSSTALK MITIGATION	136
6.7. PICO FRAMEWORK: SENSITIVITY ANALYSIS	137
6.8. EXPERIMENTS	139
6.8.1. EXPERIMENTAL SETUP	139
6.8.2. EXPERIMENTAL RESULTS WITH CORONA PNOC.....	140
6.9. CONCLUSIONS.....	144
7. HYDRA: HETERODYNE CROSSTALK MITIGATION WITH DOUBLE MICRORING RESONATORS AND DATA ENCODING FOR PHOTONIC NOCS.....	145
7.1. MOTIVATION AND CONTRIBUTION.....	145
7.2. RELATED WORK	149
7.3. PV-AWARE CROSSTALK ANALYSIS.....	151
7.3.1. IMPACT OF LOCALIZED TRIMMING ON CROSSTALK.....	152
7.3.2. IMPACT OF THERMAL TUNING OF MR ON CROSSTALK.....	155
7.3.3. PV-AWARE CROSSTALK MODELS FOR CORONA PNOC	159
7.3.4. MODELING PV OF MR DEVICES IN CORONA PNOC.....	162
7.4. HYDRA FRAMEWORK: OVERVIEW	162
7.5. CROSSTALK MITIGATION WITH DMCM	164
7.5.1. MODELING OF DMR FILTERS	166
7.5.2. OVERHEAD ANALYSIS FOR OUR DMCM SCHEME.....	167
7.6. CROSSTALK MITIGATION WITH EDCM	168
7.7. HYDRA INTEGRATION WITH PNOCS.....	170

7.8. EVALUATION.....	173
7.8.1. SIMULATION SETUP.....	173
7.8.2. WORST-CASE OSNR COMPARISON FOR VARIOUS PNOCS	174
7.8.3. OVERHEAD ANALYSIS OF HYDRA WITH VARIOUS PNOCS	176
7.9. CONCLUSIONS.....	180
8. ISLANDS OF HEATERS: A NOVEL THERMAL MANAGEMENT FRAMEWORK FOR PHOTONIC NOCS.....	181
8.1. MOTIVATION AND CONTRIBUTION.....	181
8.2. ISLANDS OF HEATERS BASED DYNAMIC THERMAL MANAGEMENT (IHDTM).....	184
8.2.1. THERMAL ISLANDS	185
8.2.2. TEMPERATURE-AWARE THREAD MIGRATION SCHEME (TATM).....	188
8.2.2.1. OBJECTIVE	188
8.2.2.2. TEMPERATURE PREDICTION MODEL	190
8.2.2.3. THERMAL MANAGEMENT ALGORITHM.....	193
8.3. EXPERIMENTS, RESULTS, AND ANALYSIS	195
8.3.1. EXPERIMENT SETUP	195
8.3.2. EXPERIMENTAL RESULTS	198
8.4. CONCLUSIONS.....	203
9. LIBRA: THERMAL AND PROCESS VARIATION AWARE RELIABILITY MANAGEMENT IN PHOTONIC NETWORKS-ON-CHIP	204
9.1. INTRODUCTION.....	204
9.2. RELATED WORK	206
9.3. IMPACT OF TV AND PV ON DWDM BASED PNOCS.....	209
9.3.1. IMPACT OF TV ON DWDM BASED PNOCS	210

9.3.2. IMPACT OF PV ON DWDM BASED PNOCS	212
9.3.3. MODELING TV AND PV IN PNO ARCHITECTURES.....	213
9.4. OVERCOMING PV/TV INDUCED RESONANCE WAVELENGTH SHIFTS	217
9.5. LIBRA FRAMEWORK: OVERVIEW	220
9.6. TV AND PV VARIATION AWARE MICRORING ASSIGNMENT (TPMA)	221
9.6.1. THERMAL VARIATION AWARE MR ASSIGNMENT (TMA).....	221
9.6.2. READAPTING TMA FOR PROCESS VARIATIONS (PMA).....	225
9.7. VARIATION AWARE ANTI WAVELENGTH-SHIFT DYNAMIC THERMAL MANAGEMENT (VADTM)	228
9.7.1. OBJECTIVE.....	228
9.7.2. THERMAL MANAGEMENT FRAMEWORK.....	230
9.8. EXPERIMENTAL RESULTS	231
9.8.1. EXPERIMENT SETUP	231
9.8.2. SENSITIVITY ANALYSIS.....	233
9.8.3. COMPARISON RESULTS	235
9.9. CONCLUSIONS.....	242
10. ANALYZING VOLTAGE BIAS AND TEMPERATURE INDUCED AGING EFFECTS IN PHOTONIC INTERCONNECTS FOR MANYCORE COMPUTING.....	243
10.1. INTRODUCTION.....	243
10.2. RELATED WORK	245
10.3. TRIMMING (VOLTAGE BIAS) INDUCED MR AGING	246
10.3.1. OVERVIEW OF VOLTAGE BIAS INDUCED TRAP GENERATION IN MRS.....	246
10.3.2. TRAP GENERATION ANALYTICAL MODEL FOR MRS.....	248
10.3.3. AGING IMPACT ON MR RESONANCE WAVELENGTH AND Q-FACTOR.....	251

10.4. TEMPERATURE INDUCED MR AGING.....	254
10.5. IMPACT OF PROCESS VARIATIONS ON MR AGING	255
10.6. IMPACT OF MR VBTI AGING ON PNOCS	257
10.6.1. MR AGING ANALYSIS FOR CORONA AND CLOS PNOCS	257
10.6.2. MODELING PV OF MR DEVICES IN CORONA AND CLOS PNOCS	260
10.7. EXPERIMENTS	261
10.7.1. EXPERIMENT SETUP	261
10.7.2. EXPERIMENT RESULTS.....	262
10.8. CONCLUSIONS.....	265
11. SOTERIA: EXPLOITING PROCESS VARIATIONS TO ENHANCE HARDWARE SECURITY WITH PHOTONIC NOC ARCHITECTURES	266
11.1. INTRODUCTION.....	266
11.2. RELATED WORK	269
11.3. HARDWARE SECURITY CONCERNS IN PNOCS.....	270
11.3.1. DEVICE-LEVEL SECURITY CONCERNS	270
11.3.2. LINK-LEVEL SECURITY CONCERNS	271
11.4. SOTERIA FRAMEWORK: OVERVIEW.....	274
11.5. PV-BASED SECURITY ENHANCEMENT	274
11.6. RESERVATION-ASSISTED SECURITY ENHANCEMENT.....	278
11.7. IMPLEMENTING SOTERIA FRAMEWORK ON PNOCS	281
11.8. EXPERIMENTS	283
11.8.1. EXPERIMENT SETUP	283
11.8.2. OVERHEAD ANALYSIS OF SOTERIA ON PNOCS	284
11.8.3. ANALYSIS OF OVERHEAD SENSITIVITY	287

11.8.4. SUMMARY OF RESULTS AND OBSERVATIONS	288
11.9. CONCLUSIONS.....	289
12. CONCLUSION AND FUTURE WORK SUGGESTIONS.....	290
12.1. RESEARCH CONCLUSION	290
12.2. SUGGESTION FOR FUTURE WORKS.....	293
BIBLIOGRAPHY.....	294

LIST OF TABLES

Table 1 CMP micro-architecture configuration	23
Table 2 Memory intensity classification of PARSEC benchmarks	36
Table 3 Energy and losses for photonic devices [73], [74].....	37
Table 4 Photonic hardware comparison.....	37
Table 5 Properties of various PNoC Architectures	38
Table 6 Micro-Architectural Parameters for MSNoC Cluster	64
Table 7 Big Data application benchmarks, with three variants each, based on their master-servant requirements.....	80
Table 8 Energy and Losses for Photonic Devices [73], [74], [100]	81
Table 9 Photonic Hardware Comparison	89
Table 10 Notations for photonic power-loss, crosstalk-coefficients and model-parameters [100]	96
Table 11 Code words for encoding techniques.....	100
Table 12 Photonic power loss and crosstalk coefficients [100].....	116
Table 13 Worst-case OSNR results for Corona and Firefly architectures.....	116
Table 14 Notations for photonic power loss, crosstalk coefficients [100]	131
Table 15 Other model parameter notations.....	131
Table 16 Photonic power loss, crosstalk coefficients [74], [100].....	160
Table 17 Other model parameter notations [74].....	160
Table 18 Code words for EDCM technique	168
Table 19 List of TATM parameters and their definitions.....	188

Table 20 Properties of materials used by 3D-ICE tool [130], [143].....	196
Table 21 Properties of materials used by 3D-ICE Tool [130], [143].....	214
Table 22 List of VADTM parameters and their definitions	229
Table 23 Notations for photonic power loss and model parameters [28].....	257

LIST OF FIGURES

Figure 1 (a) Intel Xeon Phi 72 core CMP [5] (b) Mellanox 72 core CMP [4] with electrical NoC for inter-core communication.	2
Figure 2 Overview of photonic link with wavelength division multiplexing.....	3
Figure 3 (a) Longitudinal cross-section of photonic waveguides (b) microring resonator used in PNoC architectures.	4
Figure 4 MR acting as a (a) active modulator to remove its resonance wavelength (b) detector to detect its resonance wavelength.....	5
Figure 5 (a) trans impedance amplifier (TIA) (b)1×2 splitter (c) 2×1 combiner used in PNoC architectures.	7
Figure 6 Outline of Contributions of this dissertation	12
Figure 7 Layout of MWMR crossbar used in UltraNoC and SwiftNoC along with the arrangement of cores and their respective gateway interfaces.....	21
Figure 8 (a) Timing diagram of arbitration in UltraNoC, which shows distribution of arbitration (Ai), receiver prediction (Ri)and data slots (Di) across four MWMR waveguide groups (W1 – W4); (b) distribution of different slots within MWMR waveguide group W1 at time cycle 3. ...	25
Figure 9 (a) Timing diagram of arbitration in SwiftNoC, which shows distribution of arbitration (Ai), receiver selection (Ri), and data slots (Di) across four MWMR waveguide groups (W1 – W4); (b) distribution of different slots within MWMR waveguide group W1 at time cycle 3. ...	27
Figure 10 (a) Transmission of unicast data from Node N ₁ to Node N ₃₂ in SwiftNoC, which shows receiver selection wavelength λ_{36} in receiver selection slot (R Slot) of the MWMR waveguide; (b) Multicast of data from Node N ₁ to Nodes N ₁₈ , N ₂₄ , N ₂₆ , and N ₃₂ in SwiftNoC, which shows	

respective receiver selection wavelengths λ_{22} , λ_{28} , λ_{30} , and λ_{36} in receiver selection slot (R Slot) of the MWMR waveguide..... 28

Figure 11 Bandwidth transfer technique: a cluster can transfer its unused bandwidth to the next cluster by absorbing its own arbitration wavelength and releasing arbitration wavelength of next cluster..... 32

Figure 12 Energy-delay-product (EDP) comparison for SwiftNoC-8 and SwiftNoC-16 in a 64-core CMP with time interval window sizes (a) 100-10000 cycles (b) 100-1000 cycles (zoomed version of figure 6(a))..... 39

Figure 13 (a) Average throughput, (b) average latency comparison of SwiftNoC-8 and SwiftNoC-16 with UltraNoC-8, UltraNoC-16, Flexishare, Firefly, Corona, and EMesh architectures for a 64-core CMP. Results are shown for uniform random traffic..... 42

Figure 14 Energy-delay-product (EDP) comparison of SwiftNoC-8 and SwiftNoC-16 with UltraNoC-8, UltraNoC-16, Flexishare, Firefly, Corona, and EMesh architectures for a 64-core CMP. Results are shown for uniform random traffic with packet injection rate of 0.7..... 45

Figure 15 (a) Average throughput (b) average latency comparison of SwiftNoC-16 with random multicast traffic having 10% (SWIFTNoC-MCT-10), 20% (SWIFTNoC-MCT-20), 30% (SWIFTNoC-MCT-30), 40% (SWIFTNoC-MCT-40), and 50% (SWIFTNoC-MCT-50) of multicast messages for a 64-core CMP..... 47

Figure 16 Energy-delay-product (EDP) comparison of SwiftNoC-16-MCT-10, SwiftNoC-16-MCT-20, SwiftNoC-16-MCT-30, SwiftNoC-16-MCT-40, and SwiftNoC-16-MCT-50 for a 64-core CMP. Results are shown for uniform random traffic with different percentages of multicast traffic at packet injection rate of 0.95. 47

Figure 17 (a) Average throughput (b) average packet latency (c) average energy-per-bit (EPB) comparison of SwiftNoC-8 and SwiftNoC-16 with other architectures for a 64-core CMP. Results are shown for multi-application PARSEC workloads. 50

Figure 18 (a) Average throughput (b) average packet latency (c) average EPB comparison of SwiftNoC-8, SwiftNoC-16, and SwiftNoC-32 with other architectures for a 256-core CMP. Results are shown for multi-application PARSEC workloads. 53

Figure 19 MapReduce (a) multicast phase, (b) shuffle phase, and (c) aggregation phase of communication while executing iterative machine learning algorithms for large-scale data analytics applications. 58

Figure 20 (a) MSNoC layout with SWMR, MWSR, and power waveguides (b) master gateway interface (MGI) (c) servant gateway interface (SGI). 62

Figure 21 Distribution of reservation cycle and data cycle slots within SWMR waveguide to enable MN-to-SN communication. 65

Figure 22 (a) Transmission of unicast data from an MN to SN₈ in MSNoC, which shows receiver selection wavelength λ_8 in RCS of the SWMR waveguide; (b) Multicast of data from an MN to multiple SNs SN₈, SN₁₀, SN₁₂, and SN₁₅ in MSNoC, which shows respective receiver selection wavelengths λ_8 , λ_{10} , λ_{12} , and λ_{15} in RCS of the SWMR waveguide. 68

Figure 23 Variation of average packet latency in MSNoC cluster with (a) 32 nodes (b) 16 nodes, and (c) 8 nodes having different MWSR waveguide groups (each group has 4 waveguides) across three big data applications. 72

Figure 24 (a) Homogeneous BiGNoC with four uniform clusters C₀, C₁, C₂, C₃, with each cluster having 16 nodes, (b) Heterogeneous BiGNoC with four clusters C₀, C₁, C₂, and C₃ having 32, 16, 8, and 8 nodes, respectively. 74

Figure 25 Average packet latency comparison for (a) BiGNoC-HOM and (b) BiGNoC-HET in a 256-core CMP with different buffer depths (8-40).....	83
Figure 26 (a) Normalized throughput, (b) normalized EDP comparison of BiGNoC-HOM with BiGNoC-HET for 256-core CMP. Results are shown for multi-application workloads and normalized w.r.t. BiGNoC-HET.....	86
Figure 27 Normalized (a) throughput (b) latency (c) EPB comparison of BiGNoC-HET with other architectures for a 256-core CMP. Results are for multi-application workloads and normalized w.r.t. EMesh.....	88
Figure 28 MR operation phases in DWDM-based waveguides (a) modulator modulating in resonance-wavelength (b) modulator in passing (through) mode (c) detector in passing-mode (d) detector in detecting-mode.....	93
Figure 29 Detector-wise signal power-loss, crosstalk-noise power-loss, and minimum optical-OSNR in worst-case power-loss node for Corona (a) baseline with 64-detectors (b) PCTM5B with 65-detectors (c) PCTM6B with 66-detectors.....	103
Figure 30 (a) Normalized-latency and normalized energy-delay-product (EDP) comparison between Corona baseline and Corona with PCTM5B and PCTM6B, for PARSEC benchmarks. Results are normalized to the baseline Corona results; (b) Worst-case OSNR (on-top), normalized average-latency (bottom-left) and EDP (bottom-right) for PARSEC benchmarks running on the baseline Firefly architecture and Firefly with PCTM5B and PCTM6B.....	107
Figure 31 Transmission spectrum of the cascaded microring modulators when using (a) smaller wavelength spacing (b) larger wavelength spacing.....	111

Figure 32 WSP technique: variable WSP-node increases wavelength spacing by 100% from λ to 2λ in the bottom data waveguide of the PNoC and the modulating node on the waveguide modulates on available wavelengths.....	115
Figure 33 Detector-wise signal power loss, crosstalk noise power loss and minimum OSNR in MPLN for Corona (a) baseline with 64-detectors (b) WSP increased by 20% with 53-detectors (c) WSP increased by 40% with 46-detectors (d) WSP increased by 60% with 40-detectors (e) WSP increased by 80% with 36-detectors (f) WSP increased by 100% (doubled) with 32-detectors.	117
Figure 34 (a) Throughput, and (b) energy-delay product (EDP) comparison between Corona baseline and Corona configurations with WSP_20%, WSP_40%, WSP_60%, WSP_80% and WSP_100%, for PARSEC suite.....	120
Figure 35 (a) Throughput, and (b) energy-delay product (EDP) comparison between Firefly baseline and Firefly configurations with WSP_20%, WSP_40%, WSP_60%, WSP_80% and WSP_100%, for PARSEC suite.....	121
Figure 36 Impact of PV-induced resonance shifts on MR operation in DWDM-based waveguides (note: only PV-induced red resonance shifts are shown): (a) MR as active modulator modulating in resonance wavelength with PV-induced red resonance shifts (b) MR as active detector detecting its resonance wavelength with PV-induced red shifts.	127
Figure 37 Transmission spectrum of MR groups with (a) high channel gap (CG) (b) low channel gap (CG); (C) IMCM at low channel gap.....	134
Figure 38 Overview of proposed PVCM technique	136
Figure 39 Sensitivity analysis in terms of worst-case OSNR for Corona PNoC with PICO allowing 0%, 25%, 50% and 100% ratio of shield bits to data bits across 100 process variation maps; average power consumption for each configuration is also shown on the top of each bar.	138

Figure 40 Worst-case OSNR comparison of PICO with PCTM5B [28] and PCTM6B [28] for Corona PNoC considering 100 process variation maps..... 141

Figure 41 (a) normalized latency and (b) energy-delay product (EDP) comparison between Corona baseline and Corona with PCTM5B, PCTM6B, and PICO techniques, for PARSEC benchmarks. Latency results are normalized to the baseline Corona architecture results. 142

Figure 42 Impact of PV-induced resonance shifts on MR operation in DWDM waveguides (note: only PV-induced red resonance shifts are shown): (a) MR as active modulator with PV-induced red shift, modulating in-resonance wavelength (b) detector-coupled MR filter with PV-induced red shift, filtering its resonance wavelength and dropping it on the detector. 151

Figure 43 (a) Effect of localized trimming, (b) effect of thermal tuning, on the Q-factor and fractional increase in coupling factor of an example MR. Here, the fractional increase in coupling factor is calculated w.r.t. the original coupling factor of the MR without PV..... 158

Figure 44 Overview of cross-layer HYDRA framework that integrates a device-level IM-aware crosstalk mitigation mechanism (IMCM) (see chapter 6), a device-level double MR based crosstalk mitigation mechanism (DMCM) and a circuit-level 5-bit crosstalk mitigation mechanism (EDCM). 163

Figure 45 Coupling factor (ϕ/ϕ') variation with increase in gap between the non-resonant wavelength available in the photonic waveguide and the resonance wavelength of (a) a single MR filter and (b) a DMR filter. 164

Figure 46 Crosstalk mitigation with double microring resonators: (a) MR detector operation when receiving its resonance wavelength; (b) double MR operation when receiving its resonance wavelength. 165

Figure 47 Organization of MR and DMR detectors in a detecting node on a photonic data waveguide with the EDCM mechanism.	167
Figure 48 Worst-case OSNR comparison of HYDRA with PCTM5B [28], PCTM6B [28], and PICO [31] for Corona, Firefly, and Flexishare PNoCs. Bars show mean values of worst-case OSNR across 100 PV maps; confidence intervals show variation in worst-case OSNR.	175
Figure 49 (a) Normalized average latency and (b) energy-delay product (EDP) comparison between Corona baseline and Corona configurations with PCTM5B, PCTM6B, PICO, and HYDRA techniques, for PARSEC benchmarks. Latency results are normalized to the baseline Corona results. In the EDP plot, bars represent mean values of EDP across 100 PV maps; confidence intervals show variation in EDP.	177
Figure 50 (a) Normalized average latency and (b) energy-delay product (EDP) comparison between different variants of Firefly and Flexishare PNoCs which include their baselines and their variants with PCTM5B, PCTM6B, PICO, and HYDRA techniques, for PARSEC benchmark applications. Latency results are normalized with their respective baseline architecture results. Bars represent mean values of average latency and EDP for 100 PV maps; confidence intervals show variation in average latency and EDP across PARSEC benchmarks.	179
Figure 51 Impact of thermal variations on MRs.	183
Figure 52 Peak thermal gradient (in Kelvin) across a 64-core chip running 48-threaded PARSEC [43] and SPLASH-2 [131] benchmarks.	183
Figure 53 IHDTM framework with device-level thermal islands and system-level temperature-aware thread migration mechanism (TATM).	184
Figure 54 (a) MR with adaptive heater (b) Thermal tuning of MR.	186
Figure 55 Non-linear support vector based regression prediction model.	189

Figure 56 Actual and predicted maximum temperature variation with execution time for (a) fluidanimate (FA) and (b) radiosity (RD) benchmarks run on a 64-core platform executing 32-threads.	192
Figure 57 Overview of TATM technique with support vector regression (SVR) based temperature prediction model.	193
Figure 58 Maximum temperature comparison of IHDTM with RATM and PDTM for (a) 48 and (b) 32 threaded PARSEC and SPLASH-2 benchmarks executed on 64-core CMP with Corona PNoC.	197
Figure 59 Normalized power (Laser Power (LP), Trimming and tuning power (TP) and modulating and detecting Power (MDP)) comparison of IHDTM with RATM and PDTM for (a) 48 and (b) 32 threaded applications of PARSEC and SPLASH-2 suites executed on Corona PNoC architectures for a 64-core multicore system. Results shown are normalized w.r.t RATM.	199
Figure 60 Normalized average power (laser power (LP), trimming and tuning power (TP) and modulating and detecting power (MDP)) comparison of IHDTM with RATM and PDTM for (a) 48 and (b) 32 threaded applications of PARSEC and SPLASH-2 suites executed on Flexishare PNoC for a 64-core system. Power results are normalized wrt RATM results. Bars represent mean values of power dissipation; confidence intervals show variation in power across PARSEC and SPLASH-2 benchmarks.	200
Figure 61 Normalized execution time comparison of IHDTM with RATM and PDTM for (a) 48 and (b) 32 threaded applications of PARSEC and SPLASH-2 suites executed on Corona PNoC for a 64-core system. Results shown are normalized w.r.t RATM.	201
Figure 62 Normalized average execution time comparison of IHDTM with RATM and PDTM for Flexishare PNoC running (a) 48; and (b) 32 threaded applications from PARSEC and SPLASH-2	

suites executed on 64-core system. Results are normalized wrt RATM results. Bars represent mean values of execution time; confidence intervals show variation in execution time across PARSEC and SPLASH-2 benchmarks. 202

Figure 63 Impact of temperature increase on an MR bank..... 209

Figure 64 Impact of PV on DWDM based PNoCs 212

Figure 65 Simulation framework to analyze TV and PV in a manycore system with a PNoC architectures; the framework integrates performance, power, thermal, and variation simulators..... 214

Figure 66 (a) spatial variation in peak temperatures (b) histogram of peak TV-induced resonance wavelength variation across a chip of size 400mm^2 using 3D ICE tool while executing 64 threaded PARSEC and SPLASH2 benchmark applications on a 64-core CMP. 216

Figure 67 (a) PV-induced resonance wavelength variation (b) histogram of resonance wavelength variation across a chip of size 400mm^2 217

Figure 68 Periodic resonances (R_1 - R_4) of an example bank of four MRs and their assigned carrier wavelengths (λ_1 - λ_4) for (a) an ideal case with no resonance shifts, (b) a case with systematic blue-shifts in resonances, (c) a case with random red-shifts in resonances. 219

Figure 69 Overview of LIBRA framework that integrates a device-level thermal and process variation aware microring assignment mechanism (TPMA) and a system-level variation aware anti wavelength-shift dynamic thermal management (VADTM) technique. 221

Figure 70 Red shift of MR with increase in temperature from IRTs T_i to T_{i+1} with trimming and tuning range of temperatures between these IRTs. 222

Figure 71 Thermal aware assignment of microrings (R_{1-n}) to wavelengths (λ_{1-n}) at four successive IRTs T_1 , T_2 , T_3 , and T_4 in TMA mechanism..... 223

Figure 72 Impact of PV-induced red and blue shift on boundary temperature on TMA. 225

Figure 73 Boundary temperature adaptation for larger PV-induced blue shifts in PMA..... 227

Figure 74 Overview of VADTM in LIBRA framework with support vector regression (SVR) based temperature prediction model. 229

Figure 75 Percentage of decrease in trimming/tuning power (TP) and percentage of increase in execution time (ET) comparison across different ΔZ_{tu} values for LIBRA framework implemented on Flexishare PNoC in a 64-core CMP executing blackscholes (BS), Facesim (FS), and Fluidanimate (FA). Presented results are averaged across 100 PV maps. All percentage increments/decrements are calculated w.r.t baseline Flexishare PNoC employing frequency align scheduling policy (FATM). 233

Figure 76 Maximum temperature comparison for LIBRA with RATM [133], FATM [145], PDTM [139] and SPECTRA [33], for (a) 48 thread, and (b) 32 thread PARSEC and SPLASH-2 benchmarks executing on 64-core manycore system with Corona PNoC. Bars show mean values of maximum temperature across 100 PV maps; confidence intervals show variation in maximum temperature. 235

Figure 77 Normalized power dissipation (Laser Power, Dithering Power, Trimming/Tuning power, and Modulating and Detecting (Tx/Rx) Power) comparison for LIBRA with RATM [133], FATM [145], PDTM [139] and SPECTRA [33] for 48 threaded applications of PARSEC and SPLASH-2 suites executed on (a) Corona (b) Flexishare PNoC architectures for a 64-core manycore system. Results shown are normalized w.r.t RATM, therefore, RATM does not have confidence intervals. Bars show mean values of power dissipation across 100 PV maps; confidence intervals show variation in power dissipation. 238

Figure 78 Normalized average execution time comparison of LIBRA with RATM [133], FATM [145], PDTM [139] and SPECTRA [33] for (a) Corona; and (b) Flexishare PNoCs for 48 threaded applications from PARSEC and SPLASH-2 suites executed on 64-core system. Results shown are normalized wrt RATM. 239

Figure 79 Normalized energy consumption comparison of LIBRA with RATM [133], FATM [145], PDTM [139] and SPECTRA [33] for (a) Corona; and (b) Flexishare PNoCs for 48 threaded applications from PARSEC and SPLASH-2 suites executed on a 64-core system. Results shown are normalized wrt RATM, therefore, RATM does not have confidence intervals. Bars show mean values of energy consumption across 100 PV maps; confidence intervals show variation in energy consumption. 241

Figure 80 Cross-section of a tunable MR with PN junction in its core to facilitate carrier injection into and removal from core with voltage biasing. 247

Figure 81 Distribution of electric field (E) across (a) MR waveguide; (b) Si-SiO₂ boundary B2 when -4V bias voltage is applied across PN junction. 247

Figure 82 (a) Microring resonator 3D-view with Si-core, SiO₂-cladding, and metal contacts for voltage biasing; (b) top view of MR which shows hydrogen diffusion length (λ_D) across its cladding. 250

Figure 83 Variation of resonance wavelength red shift ($\Delta\lambda_{RWRS}$) and Q_A with operation time at three operating temperatures 300K, 350K, and 400K. 255

Figure 84 Variation of Q_A and resonance wavelength red shift ($\Delta\lambda_{RWRS}$) with operation time at four bias voltages -2V, -4V, -6V, and -8V. 256

Figure 85 Worst-case signal power loss analysis of (a) Corona PNoC and (b) Clos PNoC, with 1 Year, 3 Years, and 5 Years of aging across 100 PV maps. 262

Figure 86 EDP comparison of (a) Corona and (b) Clos PNoCs with 1 Year, 3 Years, and 5 Years of aging considering 100 process variation maps.	263
Figure 87 Impact of (a) malicious modulator MR, (b) malicious detector MR on data in DWDM-based photonic waveguides.....	271
Figure 88 Impact of (a) malicious modulator (source) bank, (b) malicious detector bank on data in DWDM-based photonic waveguides.	272
Figure 89 Overview of proposed SOTERIA framework that integrates a circuit-level PV-based security enhancement (PVSC) scheme and an architecture-level reservation-assisted security enhancement (RVSC) scheme.....	273
Figure 90 Overview of proposed PV-based security enhancement scheme.	277
Figure 91 Reservation-assisted data transmission in DWDM-based photonic waveguides (a) without RVSC, (b) with RVSC.....	279
Figure 92 Comparison of (a) worst-case signal loss and (b) laser power dissipation of SOTERIA framework on Firefly and Flexishare PNoCs with their respective baselines considering 100 process variation maps.....	284
Figure 93 (a) normalized average latency and (b) energy-delay product (EDP) comparison between different variants of Firefly and Flexishare PNoCs that include their baselines and their variant with SOTERIA framework, for PARSEC benchmarks. Latency results are normalized with their respective baseline architecture results.	286
Figure 94 (a) normalized latency and (b) energy-delay product (EDP) comparison between Flexishare baseline and Flexishare with 4, 8, 16, and 24 SOTERIA enhanced MWMMR waveguide groups, for PARSEC benchmarks. Latency results are normalized to the baseline Flexishare results.	288

LIST OF ALGORITHMS

Algorithm 1 Application scheduling in BiGNoC	77
Algorithm 2 Thermal management of MR	186
Algorithm 3 TATM thread migration algorithm	194
Algorithm 4 VADTM thread migration algorithm	230

LIST OF RESEARCH PUBLICATIONS

- **S. V. R. Chittamuru**, S. Desai, and S. Pasricha, “A Reconfigurable Silicon-Photonic Network with Improved Channel Sharing for Multicore Architectures,” ACM Great Lakes Symposium on VLSI, May 2015. (**Best Paper Award**)
- **S. V. R. Chittamuru**, S. Pasricha, “Crosstalk Mitigation for High-Radix and Low-Diameter Photonic NoC Architectures”, IEEE Design and Test (D&T), vol.32, no.3, pp.29-39, June 2015.
- **S. V. R. Chittamuru**, S. Pasricha, “Improving Crosstalk Resilience with Wavelength Spacing in Photonic Crossbar-based Network-on-Chip Architectures,” IEEE Midwest Symposium on Circuits and Systems (MWSCAS), Aug. 2015.
- **S. V. R. Chittamuru**, S. Pasricha, “SPECTRA: A Framework for Thermal Reliability Management in Silicon-Photonic Networks-on-Chip,” IEEE International Conference on VLSI Design (VLSID), Jan. 2016.
- **S. V. R. Chittamuru**, I. Thakkar, S. Pasricha, “Process Variation Aware Crosstalk Mitigation for DWDM based Photonic NoC Architectures,” IEEE International Symposium on Quality Electronic Design (ISQED), Mar. 2016. (**Best Paper Award Candidate**)
- **S. V. R. Chittamuru**, I. Thakkar, S. Pasricha, “PICO: Mitigating Heterodyne Crosstalk Due to Process Variations and Intermodulation Effects in Photonic NoCs,” IEEE/ACM Design Automation Conference (DAC), June 2016.

- I. Thakkar, **S. V. R. Chittamuru**, and S. Pasricha, “A Comparative Analysis of Front-End and Back-End Compatible Silicon Photonic On-Chip Interconnects,” IEEE/ACM International Workshop on System-Level Interconnect Prediction (SLIP), June 2016. **(Best Paper Award)**
- I. Thakkar, **S. V. R. Chittamuru**, S. Pasricha, “Run-Time Laser Power Management in Photonic NoCs with On-Chip Semiconductor Optical Amplifiers,” IEEE/ACM International Symposium on Networks-on-Chip (NOCS), Aug. 2016.
- I. Thakkar, **S. V. R. Chittamuru**, and S. Pasricha, “Mitigation of Homodyne Crosstalk Noise in Silicon Photonic NoC Architectures with Tunable Decoupling,” in ACM/IEEE International Conference on Hardware/Software Codesign and System Synthesis (CODES+ISSS), Oct. 2016.
- **S. V. R. Chittamuru**, D. Dang, R. Mahapatra, and S. Pasricha, “Islands of Heaters: A Novel Thermal Management Framework for Photonic NoCs,” in IEEE/ACM Asia and South Pacific Design Automation Conference (ASPDAC), Jan. 2017.
- **S. V. R. Chittamuru**, S. Desai, and S. Pasricha, "SWIFTNoC: A Reconfigurable Silicon-Photonic Network with Multicast Enabled Channel Sharing for Multicore Architectures," in ACM Journal on Emerging Technologies in Computing Systems (JETC), vol. 13, no. 58, Feb. 2017.
- **S. V. R. Chittamuru**, I. Thakkar, and S. Pasricha, "Analyzing Voltage Bias and Temperature Induced Aging Effects in Photonic Interconnects for Manycore Computing," in International Workshop on System-Level Interconnect Prediction (SLIP), June. 2017.

- I. Thakkar, **S. V. R. Chittamuru**, and S. Pasricha, "Improving the Reliability and Energy-Efficiency of High-Bandwidth Photonic NoC Architectures with Multilevel Signaling," in IEEE/ACM International Symposium on Networks-on-Chip (NOCS), Oct. 2017.
- **S. V. R. Chittamuru**, I. Thakkar, S. Pasricha, "HYDRA: Heterodyne Crosstalk Mitigation with Double Microring Resonators and Data Encoding for Photonic NoCs," in IEEE Transactions on VLSI Systems (TVLSI), vol. 26, no. 1, Jan. 2018.
- **S. V. R. Chittamuru**, I. Thakkar, S. Pasricha, "LIBRA: Thermal and Process Variation Aware Reliability Management in Photonic Networks-on-Chip," in IEEE Transactions on Multi-Scale Computing Systems (TMSCS). (*Under review*)
- **S. V. R. Chittamuru**, D. Dang, S. Pasricha, and R. Mahapatra, "BiGNoC: Accelerating Big Data Computing with Application-Specific Photonic Network-on-Chip Architectures," in IEEE Transactions on Parallel and Distributed Systems (TPDS). (*Under review*)
- **S. V. R. Chittamuru**, I. Thakkar, V. Bhat, and S. Pasricha, "SOTERIA: Exploiting Process Variations to Enhance Hardware Security with Photonic NoC Architectures," in IEEE/ACM Design Automation Conference (DAC), June 2018. (*Under review*)

1. INTRODUCTION

Modern chip manycore processor (CMP) design aims to meet the rapidly growing performance demands of modern applications with minimum power dissipation. This chapter outlines the design challenges of CMPs, and also emphasizes the importance of on-chip communication in CMPs. Furthermore, this chapter motivates the usage of photonics-based Network-on-Chip (NoC) architectures for communication in future CMPs as they can enable higher bandwidth and lower dynamic (or data-dependent) power dissipation compared to traditional electrical NoCs. In addition, this chapter also presents the challenges of performance, reliability, static (or non-data dependent) power dissipation, and security in the design of Photonic Network-on-Chip (PNoC) architectures, and presents an outline of a cross-layer framework that addresses these challenges.

1.1. MOTIVATION FOR CMP DESIGN

In the era of cloud computing and the internet-of-things (IoT), modern applications have higher performance requirements. Advances in technology scaling over the past several decades have enabled the integration of billions of transistors on a single die. Such a massive number of transistors has allowed multiple processing cores and more memory to be integrated on a chip, allowing new chip manycore processors (CMPs) [1] to meet the rapidly growing performance demands of modern applications with lower power dissipation. An efficient on-chip communication fabric is essential to satisfy communication bandwidth and latency constraints of these CMPs. It is therefore becoming evident that focus on communication architecture design, customization, and exploration can provide huge performance gains in CMPs. Most processing

systems that include fewer processors encompass a hierarchical or crossbar-type, bus-based communication fabric. However, as the number of on-chip cores increases, bus-based communication architectures do not scale well in terms of bandwidth, clocking frequency, and power dissipation [1] because they are more susceptible to ultra-deep submicron (UDSM) effects [2] such as high signal propagation latency and high crosstalk. NoCs are now considered viable options for homogeneous CMPs as well as application specific and heterogeneous multi-processor systems-on-chip (MPSoCs). NoCs offer significant benefits in bandwidth, scalability, and reliability compared to traditional hierarchical and crossbar-based shared bus communication architectures in UDSM technologies [2]. NoCs with packet-switched network fabrics and routers can transfer data between on-chip components [3] [4] at very high data rates. Therefore, manycore processor designs have shifted toward using NoC communication fabrics instead of shared buses. Two contemporary CMPs with electrical NoCs are presented in Figure 1.

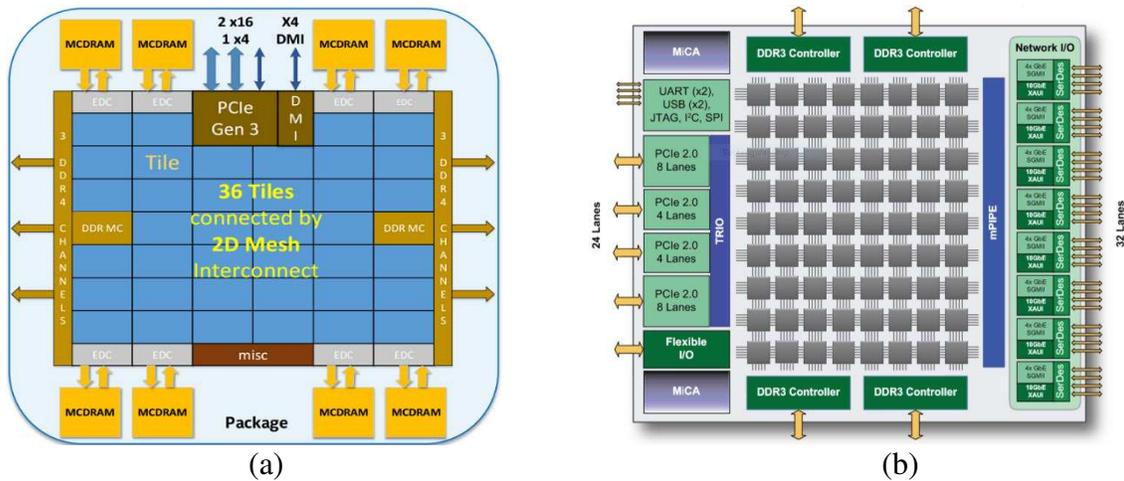


Figure 1 (a) Intel Xeon Phi 72 core CMP [5] (b) Mellanox 72 core CMP [4] with electrical NoC for inter-core communication.

As core counts continue to steadily increase, electrical NoC communication fabrics [4] [6] [5] are beginning to suffer from cripplingly high power dissipation and severely reduced

performance [7]. Moreover, the susceptibility of metallic interconnects to crosstalk and electromagnetic interference has also increased with technology scaling, which has further reduced the performance and reliability of electrical NoCs [2]. Therefore, there is a crucial need to investigate new and more viable alternatives to metallic interconnects for NoCs.

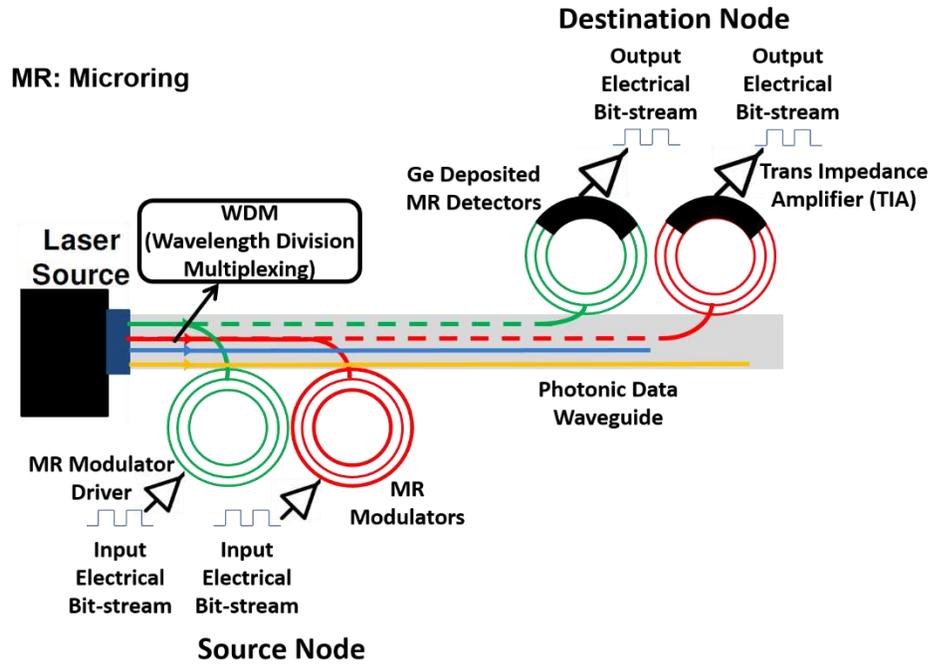


Figure 2 Overview of photonic link with wavelength division multiplexing

1.2. PHOTONIC INTERCONNECTS

Recent advances in the area of silicon nanophotonics have enabled the integration of photonic devices with CMOS circuits. The resulting on-chip photonic interconnects (shown in Figure 2) have demonstrated several prolific advantages over their metallic counterparts. Photonic interconnects enable near light speed transfers as they employ photons for data communication which are $10\times$ faster than the electrons in metallic (copper) interconnects [8]. Photonic links can also achieve distance-independent bit-rates unlike the distance dependent (crosstalk-limited) lower

bit-rates in electrical wires. The photonic links are also able to employ dense wavelength division multiplexing (DWDM) [9] to achieve a bandwidth density that is $5\times$ higher than that achieved by electrical wires. In DWDM-based photonic communication, multiple wavelengths of light can be used to simultaneously transfer multiple streams of data in a single photonic waveguide as shown in Figure 2. Additionally, because photonic links dissipate energy only at the endpoints of the communication channel [8] with low crosstalk [7] they have lower dynamic (or data-dependent) power dissipation (about 7.9 fJ/bit) than that of electronic links. Thus silicon nanophotonics is being considered as an exciting new option for integration in future NoCs. Several photonic devices such as Microring Resonators (MRs), waveguides, and photodetectors have already been successfully fabricated and demonstrated at the chip level [10]. These devices have been used as a foundation for several PNoC architectures [11], [12], [13], [14].



Figure 3 (a) Longitudinal cross-section of photonic waveguides (b) microring resonator used in PNoC architectures.

1.2.1. PHOTONIC WAVEGUIDES

In PNoC architectures, photonic waveguides are used to traverse optical signals from a source core to a destination core. Photonic waveguides, as shown in Figure 3(a), use a high refractive index silicon (Si) core (i.e., $n_{\text{Si}} = 3.5$) and low refractive index silicon-di-oxide (SiO₂) cladding (i.e., $n_{\text{SiO}_2} = 1.5$) fabricated on a silicon-on-insulator (SOI) platform. These waveguides have

a lower pitch and area footprint than the polymer waveguides used in [2]. Waveguides fabricated on an SOI platform have other advantages such as lower losses (on the order of 1 dB/cm) and the malleability to be curved with bend radii of $\sim 5\mu\text{m}$ [15]. Malleability of these photonic waveguides and the SOI platform's high refractive index contrast enables fabrication of compact modulators which require lower drive voltage for high frequency operation. To support high bandwidths for future CMP applications, these photonic waveguides support dense wavelength division multiplexing (DWDM) [16], with multiple wavelengths available for concurrent data transfers in each waveguide.

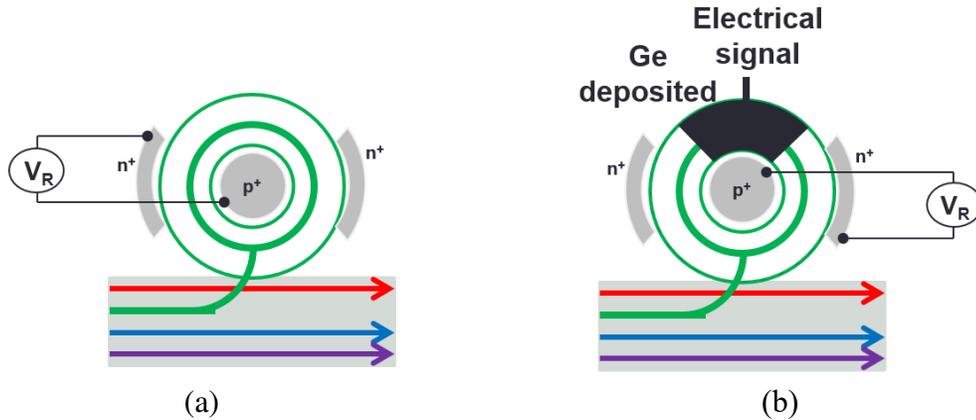


Figure 4 MR acting as a (a) active modulator to remove its resonance wavelength (b) detector to detect its resonance wavelength.

1.2.2. MICRORING RESONATORS

To transmit data between cores through a photonic waveguide, electrical to optical (E/O) conversion at the source and an optical to electrical (O/E) conversion at the destination is required. MRs can enable both E/O and O/E conversion in PNoCs. MRs modulate light for transmission of data at a source (data-modulation phase). MRs also detect light-modulated data from the waveguide at the destination (data-detection phase) and subsequently help with the generation of

proportional electrical signals that are amplified by Trans-Impedance Amplifiers (TIAs). An MR can be functionally described as a circular photonic waveguide with a small diameter as shown in Figure 3(b).

MRs are wavelength selective and couple light when the relation $\lambda \times m = n_{eff,ring} \times 2\pi R$ is satisfied, where R is the radius of the microring resonator, $n_{eff,ring}$ is the effective refractive index, m is an integer value, and λ is the resonant wavelength [17]. As resonance wavelength is a function of R and $n_{eff,ring}$, by changing R and $n_{eff,ring}$, the resonant wavelength of the MR can be altered. It is necessary to alter resonance wavelength of an MR to remove a wavelength (in active mode to write ‘0’-bit) from a data waveguide, and to let a wavelength pass through (in passive mode to write ‘1’-bit) in a data waveguide. In general, alteration in resonance wavelength of an MR by $\Delta\lambda$ is achieved with Δn_{eff} change in effective refractive index. There are two ways that can change the effective refractive index of an MR. Injection or removal of carriers (electrons) from the Si core of an MR alters its effective refractive index due to the Electro Optic (EO) effect [18]. Heating of MR’s also alters its effective refractive index due to the Thermo Optic (TO) effect [19]. More details about EO and TO effects are presented in chapter 7 and 9. However, the former method is faster and consumes lower power compared to the latter one for smaller resonance wavelength shift (i.e., <1nm) [19]. Therefore, carrier injection/removal is predominantly used to switch MRs between active and passive modes. To enable carrier injection/removal in MRs require a series of drivers. These drivers are electrical circuits which regulate carrier injection/removal rates (by altering voltage V_R shown in Figure 4) into MRs to control their resonance wavelength shifts. An MR as a modulator is shown in Figure 4(a) that removes its resonance wavelength from the data waveguide, which converts electrical signal to optical signal. Furthermore, as shown in Figure 4(b), an MR with germanium (Ge) deposited on its Si core acts as a detector to drop the corresponding

resonance wavelength from the data waveguide and convert the optical signal back to an electrical signal.

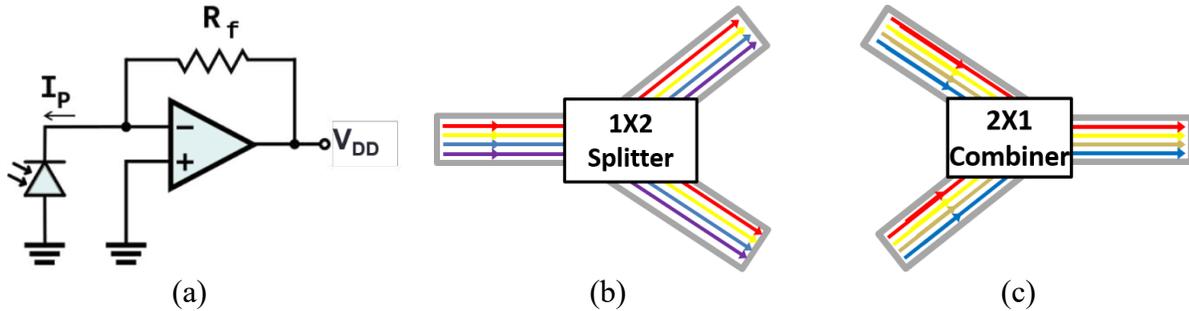


Figure 5 (a) trans impedance amplifier (TIA) (b)1x2 splitter (c) 2x1 combiner used in PNoC architectures.

1.2.3. TRANS-IMPEDANCE AMPLIFIERS, COMBINERS, AND SPLITTERS

TIAs are used to amplify detected signals at the MR detector to digital voltage levels as shown Figure 5(a). As the signals amplified by TIAs are ultimately stored and processed on the chip, their amplitudes should match the supply voltage of logic circuits (i.e., V_{DD}). To enable amplification of signals to V_{DD} , the TIAs are typically operated at 20% higher supply voltage than V_{DD} . In addition to these TIAs, PNoCs employ splitters and combiners respectively to distribute and aggregate signal power in photonic waveguides, as shown in Figure 5(b) and (c), respectively.

1.3. DESIGN CHALLENGES IN PNOCS

Despite the aforementioned advantages of high bandwidth, low latency, and low dynamic power dissipation for photonic interconnects, building PNoCs with photonic interconnects still faces several challenges. We organize these challenges into four categories: performance challenges, reliability challenges, power dissipation challenges, and security challenges.

1.3.1. PERFORMANCE CHALLENGES

Performance challenges in PNoC architecture design includes network resource contention, adaptation to application traffic, low bandwidth, and high network latency. Some prior work has given emphasis to the importance of network resource contention in photonic NoC channels and proposed arbitration techniques to resolve this contention [11] [13]. However, these approaches are limited because they do not fully exploit available network bandwidth and typically only target single parallel application workloads when designing and optimizing the proposed techniques. In emerging multicore systems where multiple applications execute simultaneously on unique subsets of cores, there is significantly greater variation in temporal and spatial characteristics of network injected traffic. For example, cores running memory intensive tasks can require more network bandwidth than cores running compute intensive tasks [20]. Furthermore, O/E and E/O conversions in the photonic NoC channel increases network latency of PNoCs.

1.3.2. RELIABILITY CHALLENGES

Reliability challenges in PNoC architecture design includes crosstalk noise, process variations, thermal variations, and aging of MRs. Crosstalk noise in MRs is classified into two types: heterodyne crosstalk noise and homodyne crosstalk noise. The homodyne crosstalk noise power of a particular wavelength affects the signal power of the same wavelength, whereas with heterodyne crosstalk the signal power gets affected by some noise power of one or more other (different) wavelengths. The strength of the heterodyne crosstalk noise at a detector MR depends on the following four attributes: (i) channel gap between the MR resonant wavelength and the adjacent wavelengths; (ii) Q-factors of neighboring detector MRs, (iii) the strengths of the non-resonant signals at the detector, and (iv) bit-rate or modulation rate of the photonic link. With an

increase in DWDM, the channel gap between two adjacent wavelengths decreases, which in turn increases heterodyne crosstalk in detector MRs. With a decrease in Q-factors of MRs, the widths of the resonant passbands of MRs increases, increasing passband overlap among neighboring MRs, which in turn increases heterodyne crosstalk. The strengths of the non-resonant signals depend on the losses faced by the non-resonant signals throughout their path from the laser source to the MR detector. When a data-modulated non-resonant signals passes by an MR, depending on its data bit-rate (modulation rate), a part of its signal power is dropped by the MR, which in turn affects the heterodyne crosstalk noise caused by these non-resonant signals.

Fabrication process variations (PV) induce variations in the width and thickness of MRs, which cause resonance wavelength shifts in MRs [21] [22]. PV-induced resonance shifts may reduce the channel gap between the resonances of the victim MRs and adjacent MRs, which increases crosstalk and worsens optical signal-to-noise-ratio (OSNR). The worsening of OSNR deteriorates the bit-error-rate (BER) in a waveguide. For example, a previous study shows that in a DWDM-based photonic interconnect, when PV-induced resonance shift is over 1/3 of the channel gap, BER increases from 10^{-12} to 10^{-6} [23]. Techniques to counteract the PV-induced resonance shifts in MRs involve realigning the resonant wavelengths by using localized trimming [18] or thermal tuning [19].

MR devices are highly sensitive to temperature fluctuations. With increase or decrease in temperature, the refractive index of an MR device changes, causing a change in its resonance wavelength. This wavelength is supposed to remain static, as the value assigned at design-time [19]. As a result of this variation in resonance wavelength, an MR may be unable to write or read data in the waveguide. As the temperature increases or decreases from the MR's design (baseline) temperature, due to the resulting variations in refractive index, each MR now resonates with a

different wavelength towards the red (i.e., red-shift) or blue (i.e., blue-shift) end of the visible spectrum. This phenomenon reduces transmission reliability and also leads to wastage of available bandwidth.

To facilitate switching of resonance-modes of an MR with voltage biasing or trimming, a PN junction is created in the Si core of the MR surrounded by SiO₂ cladding. A positive/negative voltage bias is applied to this PN-junction to inject/remove free carriers into/out of the MR's Si core. For high frequency operation and lower power consumption, an MR's PN-junction is typically operated under a negative voltage bias (or reverse bias) [24]. The application of this voltage bias generates an electric field across the MR's Si core and SiO₂ cladding boundary. Similar to MOSFETs, this electric field generates voltage bias temperature induced (VBTI) traps at the Si-SiO₂ boundary of the MR over time (i.e., VBTI aging). Our analysis has shown that these VBTI aging induced traps alter carrier concentration in the Si core of MRs, which incur resonance wavelength shifts and increase optical scattering loss in MRs to decrease their Q-factor.

1.3.3. POWER CHALLENGES

Power challenges in PNoC architecture design includes high laser power dissipation and high trimming/tuning power dissipation. Data communication with photonic signals in photonic interconnects is *lossy*. Photonic signals traversing in waveguides incur propagation and bending losses and modulators and detectors incur through losses and modulator/detector insertion losses [9]. In addition to these losses, couplers and splitters incur coupling and splitting losses. The aforementioned losses in photonic signals demand higher laser power to ensure that all the detectors along the photonic interconnect receive sufficient signal power. This laser power dissipation needs to be controlled otherwise it will reduce the energy benefits of photonic

interconnects. Another component of power dissipation in PNoC is static trimming and tuning power dissipation. As explained in subsection 1.3.2, an increase in process and thermal variations increases trimming and tuning power dissipation. Further trimming/tuning power has linear dependency on the number of MRs used within a PNoC architecture. Therefore, PNoCs with a higher number of MRs (larger photonic footprint) incur more trimming/tuning power dissipation and lead to higher energy consumption.

1.3.4. SECURITY CHALLENGES

PNoC architectures employ shared photonic waveguides to achieve higher data rates with the minimum amount of photonic hardware [11]- [13]. Several nodes in a PNoC architecture are able to read and write data on these shared waveguides. Furthermore, several PNoC architectures [11], [12], [25] send multicast or broadcast data to multiple nodes using these shared waveguides. Despite achieving higher data rates, these shared waveguides are vulnerable to security risks. A malicious node on the shared waveguide can steal or snoop the data from the shared waveguides and transmit it to a malicious core to extract sensitive information from the data. Furthermore, malicious nodes on the shared waveguides can perform deep packet inspection and inject faults on links to develop a denial-of-service (DoS) attack. In addition, malicious nodes can corrupt data on the shared waveguides and increase bit errors in PNoCs beyond correctable limits.

1.4. DISSERTATION OUTLINE

To address the challenges presented in the previous section, in this dissertation, we propose a framework for silicon photonic NoC design, with a high level preview of contributions shown in Figure 6 [25] [26] [27] [28] [29] [30] [31] [32] [33] [34] [35] [36] [37] [38] [39] [40] [41]. This

framework includes not only layer-specific solutions, but also cross-layer solutions that combine enhancements at the system-level, architecture-level, circuit-level, and device-level towards the design of reliable, energy-efficient, and secure PNoC architectures. The rest of this dissertation is organized as follows:

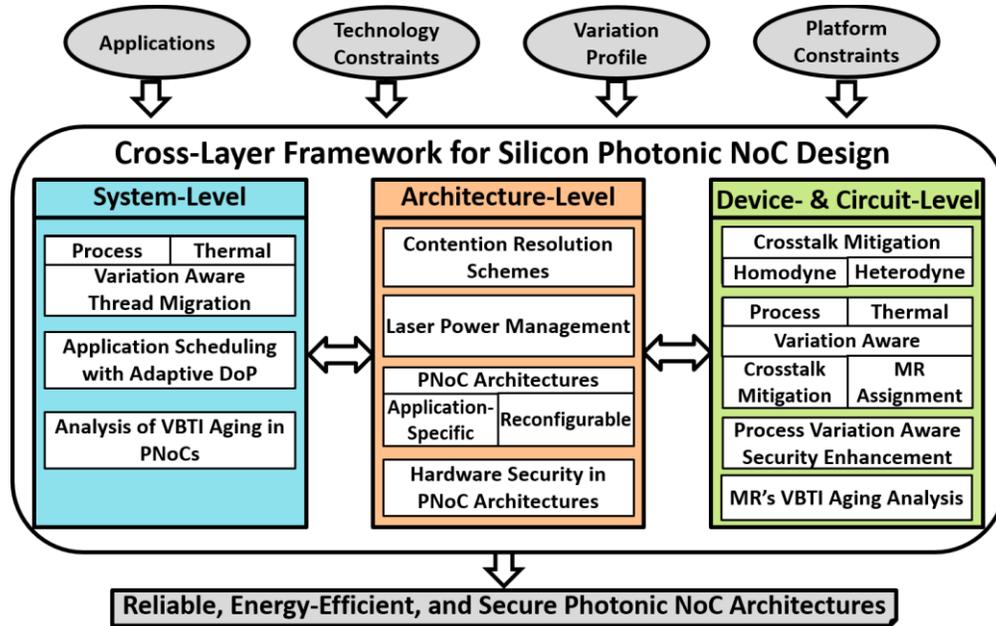


Figure 6 Outline of Contributions of this dissertation

In chapter 2, we propose a novel PNoC architecture called SwiftNoC [25] that utilizes multiple-write-multiple-read (MWMR) photonic waveguides in a crossbar topology, and supports a novel approach for dynamic performance adaptation to aggressively utilize network bandwidth and meet diverse application demands. Our SwiftNoC architecture integrates a novel distributed concurrent token stream arbitration that provides multiple simultaneous tokens and increases channel utilization in MWMR photonic waveguides. Furthermore, the SwiftNoC PNoC uses multicast enabled MWMR waveguides which facilitate the energy efficient multicast of messages.

The novel dynamic bandwidth transfer mechanism in SwiftNoC enables low-overhead transfers of unused bandwidth between clusters of cores to improve channel utilization even further. The SwiftNoC architecture also utilizes real-time monitoring of the traffic being injected into the network by different co-running applications, to facilitate dynamic arbitration wavelength injection rate modulation.

In chapter 3, we present a novel application-specific PNoC architecture for manycore chips, called BiGNoC [27], that enables the execution of large-scale data analytics applications with high throughput and ultra-low latency. We devise a master-servant cluster based communication fabric (MSNoC) with dedicated channels for master-to-servant and servant-to-master communication. Furthermore, we design a hierarchical manycore BiGNoC architecture with multiple MSNoCs to execute any combination of high performance large-scale data analytics applications. In addition, we analyze the power and performance of two variants of the BiGNoC architecture: homogenous BiGNoC (BiGNoC-HOM) and heterogeneous BiGNoC (BiGNoC-HET).

In chapter 4, we present the design of two novel circuit-level techniques that attempt to intelligently reduce crosstalk by minimizing undesirable data value occurrences in a photonic waveguide. Our first crosstalk mitigation technique uses 5-bit encoding (PCTM5B) [28] to improve the worst-case OSNR with relatively low impact on energy-delay product (EDP) for DWDM-based photonic crossbar PNoCs. Our second crosstalk-mitigation scheme with 6-bit encoding (PCTM6B) [28], more aggressively improves OSNR but with relatively higher EDP overhead. These techniques are easily implementable on any existing DWDM-based photonic crossbar without requiring major modifications to the architectures, unlike previously proposed crosstalk mitigation techniques [42] that aim to reduce crosstalk in specific PNoC architectures by

requiring modifications to their router designs. Further, our techniques are lightweight and possess low overhead.

In chapter 5, we propose novel wavelength spacing (WSP) techniques [29] to increase spacing between adjacent wavelengths in a DWDM waveguide for PNoCs. The WSP technique can help to reduce crosstalk noise and improve OSNR in DWDM-based PNoC architectures. The proposed WSP technique very effectively improves both reliability and EDP for these architectures.

In chapter 6, we present a novel crosstalk mitigation framework called *PICO* [31] to enable reliable communication in emerging PNoC-based multicore systems. *PICO* mitigates the effects of IM crosstalk by controlling signal loss of wavelengths in the waveguide and reduces trimming-induced crosstalk by intelligently reducing undesirable data value occurrences in a photonic waveguide based on the PV profile of MRs. We present device-level analytical models to capture the deleterious effects of localized trimming in MRs. Moreover, we extend this model for system-level heterodyne crosstalk analysis. Furthermore, this chapter also discusses a scheme for IM passband truncation-aware heterodyne crosstalk mitigation (IMCM) to improve worst-case OSNR of MRs by controlling non-resonant signal power. We also propose a scheme for PV-aware heterodyne crosstalk mitigation (PVCMM) to improve worst-case OSNR of detector MRs by encoding data to avoid undesirable data occurrences.

In chapter 7, we present a novel cross-layer heterodyne crosstalk mitigation framework called *HYDRA* [32] to enable reliable communication in emerging PNoC-based manycore chips. We present device-level analytical models to capture the deleterious effects of localized trimming and thermal tuning in MRs. We extend these models for system-level heterodyne crosstalk analysis. We also propose a device-level technique in this chapter for heterodyne crosstalk mitigation (DMCM) that uses double MRs to improve worst-case OSNR in detectors by tailoring the MRs'

passbands to have a steeper roll-off. Furthermore, a circuit-level technique for heterodyne crosstalk mitigation (EDCM) is proposed that aims to improve worst-case OSNR in detectors by encoding data to avoid undesirable data value occurrences. Lastly, we combine IMCM (see chapter 6), DMCM, and EDCM into a holistic cross-layer heterodyne crosstalk mitigation framework called HYDRA and evaluate it on three well-known crossbar PNoC architectures as well as prior work on heterodyne crosstalk mitigation.

In chapter 8, we propose a novel cross-layer, low-power, thermal management framework [34] that integrates an adaptive heater mechanism at the device-level and a dynamic thread migration scheme at the system-level. We present a novel temperature island framework with adaptive heater based MRs to handle thermal gradients across PNoC. Furthermore, an islands-of-heaters-based dynamic thread migration (IHDTM) scheme is also proposed in conjunction with a support-vector-regression (SVR) based temperature prediction mechanism. This scheme nullifies on-chip thermal threshold violations and also reduces trimming/tuning power for MRs.

In chapter 9, we propose a thermal and process variation aware dynamic reliability management framework called *LIBRA* [35] that integrates adaptive MR assignment at the device-level and dynamic thread migration at the system-level for PNoC-based manycore systems. The adaptive thermal and process variation aware microring assignment (TPMA) mechanism at the circuit-level tunes a set of photonic microring resonators (MRs) dynamically for reliable modulation and reception of data from a photonic waveguide in a specific temperature and process variation range. This technique aims to adapt to the changing on-chip thermal profile and maintain maximum bandwidth while minimizing trimming and tuning power in the PNoC. However, TPMA cannot control maximum on-chip temperature, whose control is critical to further minimize MR trimming and tuning power. Thus, to control maximum on-chip temperature, we devise a system-

level PV-aware anti-wavelength-drift dynamic thermal management (VADTM) scheme that uses SVR based thermal prediction and dynamic thread migration, to avoid on-chip thermal threshold violations, minimize hotspots, and reduce thermal tuning power for MRs. Both TPMA and VADTM work synergistically to reduce PNoC energy consumption.

In chapter 10, we study the VBTI aging in MRs and its impact on PNoC architectures [36]. At the device-level, we carefully developed analytical models for trap generation with VBTI aging in MRs. We also devise analytical models in this chapter that determine variations of MR resonance wavelength shifts and Q-factor with aging-induced traps. These models are further extended to examine the impact of different operating temperatures and bias voltages, as well as process variations. From those models, we follow a mathematical bottom-up approach to analyze the system-level impact of aging on different PNoC architectures.

In chapter 11, we present a framework [37] that protects data from snooping attacks and improves hardware security in PNoCs. We analyze security risks in photonic devices and extend this analysis to the link-level, to determine the impact of these risks on PNoCs. We propose a circuit-level PV-based security enhancement scheme that uses PV-based authentication signatures to protect data from snooping attacks in photonic waveguides. We propose an architecture-level reservation-assisted security enhancement scheme to improve security in DWDM-based PNoCs;

Chapter 12 concludes this dissertation. We summarize our overall body of research and make recommendations for future research.

2. SWIFTNOC: A RECONFIGURABLE SILICON-PHOTONIC NETWORK WITH MULTICAST ENABLED CHANNEL SHARING FOR MULTICORE ARCHITECTURES

With recent advances in silicon nanophotonics, photonics-based network-on-chip (NoC) architectures are being considered as a viable solution to support communication in future CMPs as they can enable higher bandwidth and lower power dissipation compared to traditional electrical NoCs. In this chapter, we present *SwiftNoC*, a novel reconfigurable silicon-photonic NoC architecture that features improved multicast-enabled channel sharing, as well as dynamic re-prioritization and exchange of bandwidth between clusters of cores running multiple applications, to increase channel utilization, and system performance. Experimental results show that *SwiftNoC* improves throughput by up to 25.4× while reducing latency by up to 72.4% and energy-per-bit by up to 95% over state-of-the-art solutions.

2.1. MOTIVATION AND CONTRIBUTION

A few prior works have emphasized the importance of network resource contention in photonic NoC channels and proposed arbitration techniques to resolve this contention [11] [13]. However, a limitation of these approaches is that they do not fully exploit available network bandwidth and typically only target single parallel application workloads when designing and optimizing the proposed techniques. In emerging multicore systems where multiple applications execute simultaneously on unique subsets of cores, there is significantly greater variation in temporal and spatial characteristics of network injected traffic. For example, cores running memory intensive tasks can require more network bandwidth than compute intensive tasks [20].

To overcome all of these shortcomings, we propose a novel photonic NoC architecture called *SwiftNoC* that utilizes multicast enabled multiple-write-multiple-read (MWMR) photonic waveguides in a crossbar topology and supports dynamic performance adaptation to aggressively utilize network bandwidth and meet diverse application demands. The *SwiftNoC* architecture improves data transfer rate in its MWMR waveguides through a better and faster arbitration mechanism. We compare *SwiftNoC* against alternative architectures with the best-known arbitration mechanisms from prior work, for synthetic and multi-threaded PARSEC [43] workloads on CMP platform sizes ranging from 64-cores to 256-cores.

The novel contributions of this chapter can be summarized as follows:

- A flexible on-chip photonic network architecture (*SwiftNoC*) that facilitates selective and reconfigurable prioritization of applications based on their time-varying performance goals;
- Multicast enabled MWMR waveguide in *SwiftNoC* that facilitates energy efficient multicast of messages;
- Improved distributed concurrent token stream arbitration that provides multiple simultaneous tokens and increases channel utilization;
- A dynamic bandwidth transfer technique with low overhead, to transfer unused bandwidth among clusters of cores;
- A mechanism to monitor traffic being injected into the network by different co-running applications, to facilitate dynamic arbitration wavelength injection rate modulation.

2.2. RELATED WORK

As per projections from the International Technology Roadmap for Semiconductors (ITRS) [44], in the near future, delay and power consumption of copper-based electrical interconnects will

become a serious bottleneck for chip design. These projections motivate the exploration of new technologies to enable viable fabrication of CMPs in future process technologies. On-chip networks in the TILE-Gx72 Processor with 72 cores [4] and Intel 80-core Terascale processor [3] consume approximately 20-30% of the total chip power. This trend is expected to continue as more wire density becomes available in future process technologies [44]. Thus the expected on-chip network power will continue to rise as we scale to several hundreds of cores on a single IC.

To overcome this challenge, several novel interconnect technologies are beginning to be explored, including carbon nanotubes (CNTs) [45] [46] [47] [48] and wireless interconnects [49] [50] [51] [52] [53] for on-chip communication. However, CNT fabrication is not yet mature and has serious practical concerns to overcome. Multiband RF transmission lines and wireless interconnects (RF-Is) require high operating frequencies in the range of hundreds of GHz to THz. Complex RF-I Frequency Division Multiple Access (FDMA), transmission lines, or on-chip antennas also entail high area, power, verification, and implementation costs. Nonetheless, these technologies are quite promising and may become more viable in the near future.

Silicon photonic on-chip interconnects are yet another promising alternative for chip level communication [54]. A considerable amount of work has focused on the design of photonic NoCs in recent years. The concept of photonic interconnects for on-chip communication was first discussed by Goodman et al. in [55]. Inter-chip photonic interconnects were explored in several works [21] [56] [57] [58] [59]. Other efforts have focused on on-chip photonic interconnects with either high-radix low-diameter photonic on-chip crossbar architectures that provide non-blocking connectivity, e.g., [11] [13] [60] [61]; or low-radix high-diameter NoCs [62] [63]. A categorization of various photonic crossbars to meet the design requirements of different CMPs is presented in [64]. Further, there was a recent effort [65] which combines reconfigurable adaptive routing and

network coding to improve power and performance in electrical NoCs which can also be extended to PNoCs. In addition to these, cross-layer solutions [33] [34] [66] were presented towards the design of thermally resilient PNoCs with enhancements at the circuit, architecture, and operating system (OS) levels. Moreover, few more cross-layer solutions [31] [40] were presented to mitigate crosstalk noise in PNoCs with enhancements at device and circuit levels.

Prior work has shown that photonic crossbars are extremely promising on-chip communication architectures to meet future on-chip bandwidths demands, but they can suffer from (i) large power dissipation, and (ii) high contention overheads for shared resources, especially when using inefficient token-based arbitration schemes. A few techniques to reduce power overhead of photonic NoCs have been proposed in literature. For example, an effective policy for runtime management of the laser source is proposed in [14]. We build on the foundations from their work to manage power dissipation in photonic crossbar NoCs in this chapter.

To reduce contention issues in crossbars, a few improved arbitration techniques have been proposed in [13] [67] that use time division multiplexing (TDM), so that a single data waveguide can be simultaneously used by more than one node in different time slots. In Flexishare [13], a token stream arbitration scheme is proposed. The scheme requires wavelengths corresponding to each data waveguide to be injected serially into different time slots of an arbitration waveguide. A node writes on the data waveguide only when it gets access to the corresponding arbitration wavelength. Subsequently, the node cannot send data again till its arbitration wavelength is injected into the arbitration waveguide, which takes N cycles for N data waveguides. The scheme leads to channel under-utilization, and performs worse as the number of nodes and waveguides increase. In [67], the token ring arbitration scheme from Corona [11] was improved with the token channel and token-slot arbitration techniques for Multiple-write-single-read (MWSR) crossbars.

Token-slot arbitration uses TDM and improves upon token channel arbitration by dividing the arbitration waveguide into fixed-size, back-to-back slots, with destination nodes circulating tokens in one-to-one correspondence to slots. A limitation of this approach is that a fixed time gap is required between two arbitration slots to set up data for transmission, which reduces the available time slots to send data. UltraNoC [26] improves upon these prior works by utilizing a more effective concurrent token stream arbitration strategy, together with support for reconfigurable cluster prioritization and bandwidth re-allocation to improve MWMR photonic channel utilization.

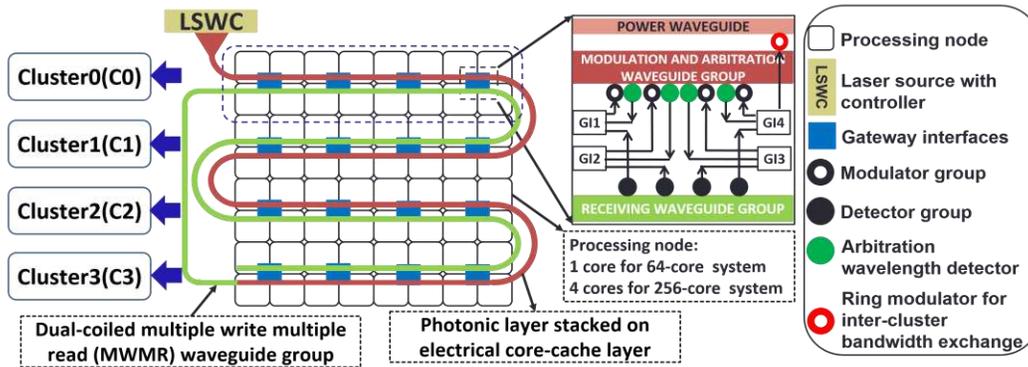


Figure 7 Layout of MWMR crossbar used in UltraNoC and SwiftNoC along with the arrangement of cores and their respective gateway interfaces.

2.3. ULTRANOC AND SWIFTNOC : PHOTONIC ARCHITECTURE OVERVIEW

2.3.1. ULTRANOC ARCHITECTURE AND TERMINOLOGY

The baseline UltraNoC architecture [26] is designed for a 64-core CMP, as shown in Figure 7. We also extend the baseline architecture to a 256-core CMP for the purposes of scalability analysis. Each core has a private L1 and shared L2 cache. In a 64-core CMP, each group of 8 cores has access to main memory via a dedicated memory controller, whereas in a 256-core CMP, each group of 16 cores has a dedicated memory controller. We have considered memory interleaving in our architecture and adapted its specific implementation from prior work [68]. A node (N) is

defined as an entity consisting of one and four cores for the 64-core and 256-core CMPs, respectively. Every node in UltraNoC is attached to a gateway interface (GI) module that facilitates transfers between the CMOS electrical layer and a photonic layer (with photonic waveguides, modulators, detectors, etc.). The entire chip is divided into four clusters (C_0, C_1, C_2, C_3) as shown in Figure 7, where a cluster contains 16 cores in a 64-core CMP and 64 cores in a 256-core CMP. Beyond 256-core CMPs (i.e. 512 or 1024 cores), we can certainly increase the number of clusters (e.g., to 8), to enable more re-configurability in these architectures. Increase in the number of clusters increases the number of arbitration wavelengths (see section 2.3.2) in the waveguide, which ultimately requires an increase in DWDM of the waveguide, incurring more power dissipation. Therefore, a careful analysis is required to remain within power constraints while meeting performance objectives. However, our scope of work is limited to 64-core and 256-core CMPs, for which four clusters provides a good trade-off between performance and power dissipation in our UltraNoC architecture. More details of the micro-architectural parameters of the cores and main memory are shown in Table 1.

A detailed layout of the UltraNoC architecture is shown in Figure 7, where 64 nodes (N_0 - N_{63}) are arranged in an 8×8 grid. Communication between cores within a node for the 256-core CMP uses an electrical 5×5 NoC router, where four of its input and output port pairs are connected to four cores and the fifth input/output port pair is connected to a GI module. A round-robin arbitration scheme is used within each node for communication between cores and the GI. For higher concentration degree (more than 4 cores within a concentrator) in a general purpose CMP platform, using a round-robin strategy is a suitable option to achieve fairness for a diverse distribution and choice of workloads; however, if workloads and task to core mapping information is available, priority based arbitration schemes (e.g., [69]) may be a better choice.

Table 1 CMP micro-architecture configuration

CMP Type	64-Core	256-Core
Number of cores	64	256
Number of clusters	4 (16 cores each)	4 (64 cores each)
Per Core:		
L1 I-Cache size/Associativity	16KB/Direct Mapped Cache	
L1 D-Cache size/Associativity	16KB/Direct Mapped Cache	
L2 Cache size/ Associativity	128KB/ Direct Mapped Cache	
L2 Coherence	MOESI	
Frequency	5 GHz	
Issue Policy	In-order	
Memory controllers	8	32
Main memory	8GB; DDR4@30ns	32GB; DDR4@30ns

Inter-node transfers are facilitated by dual-coiled MWMM waveguide groups, where each group has four MWMM waveguides. Each MWMM waveguide group in UltraNoC passes every node twice in the dual-coiled structure to enable a two pass inter-node data communication. A node has the ability to write on the first pass using its ring modulators and read from the waveguide group using its ring detectors in the second pass. As all nodes are capable of modulating (writing) in an MWMM waveguide group during the first pass, there is a need for arbitration (see Section 2.3.2 for more details) between sending nodes to ensure that the data of different senders does not destructively overlap on the shared waveguide group. Throughout this chapter this first pass portion of the waveguide group is referred to as the *modulating and arbitration waveguide group*. In the second pass of the MWMM waveguide group all nodes receive data through their respective ring detectors; hence this portion of the waveguide group is referred to as the *receiving waveguide group*. As all nodes are capable of receiving (reading) from an MWMM waveguide group during the second pass, there is a need for receiver selection (see Section 2.3.2 for more details) between receiving nodes to ensure that the designated receiver will receive data from the shared waveguide group. Further each node in our architecture is capable of sending (in the first pass) and receiving

(in the second pass) data from all the multiple MWMMR waveguide groups through their separate ring modulator and ring detector banks respectively on each individual MWMMR waveguide group. Additionally, there is a power waveguide that runs in parallel with the other waveguides, and carries arbitration wavelengths. This waveguide facilitates our bandwidth transfer and priority adaptation techniques (see Sections 2.3.3, 2.3.4).

Figure 7 depicts an expanded view of the collection of GIs for four nodes, which shows the modulating and arbitration, receiving, and power waveguide groups, along with their connection to GIs. As explained above, each modulating and arbitration, and receiving waveguide group has four MWMMR waveguides. Among these four MWMMR waveguides, the first waveguide has 68 DWDM (i.e., 68 wavelengths represented as λ_0 to λ_{67}) and the remaining three waveguides have 64 DWDM each. In the first MWMMR waveguide 4 wavelengths ($\lambda_0 - \lambda_3$) are used for arbitration and the remaining 64 wavelengths ($\lambda_4 - \lambda_{67}$) are used for data transfer and receiver selection. During the receiver selection process, each of the 64 wavelengths is assigned to a unique receiving node (i.e., $\lambda_4, \lambda_5, \dots, \lambda_{67}$ are assigned to N_0, N_1, \dots, N_{63} respectively), such that whenever a receiver detects its corresponding wavelength during a clock cycle, it switches its detectors “on” to receive data in the next clock cycle. All other receivers keep their detectors turned off to save power. More details about the usage of these wavelengths in the first MWMMR waveguide is presented in the next subsection (Section 2.3.2). As each waveguide in this MWMMR waveguide group uses 64 wavelengths for data transfer, each waveguide group in the UltraNoC architecture facilitates simultaneous transfer of a total of 512 bits of data with data modulation at both clock edges in a clock cycle. Thus in an MWMMR waveguide group, each ring modulator and detector group has 256 ring modulators and 256 ring detectors, respectively that are accessed at the positive and negative edges of the clock.

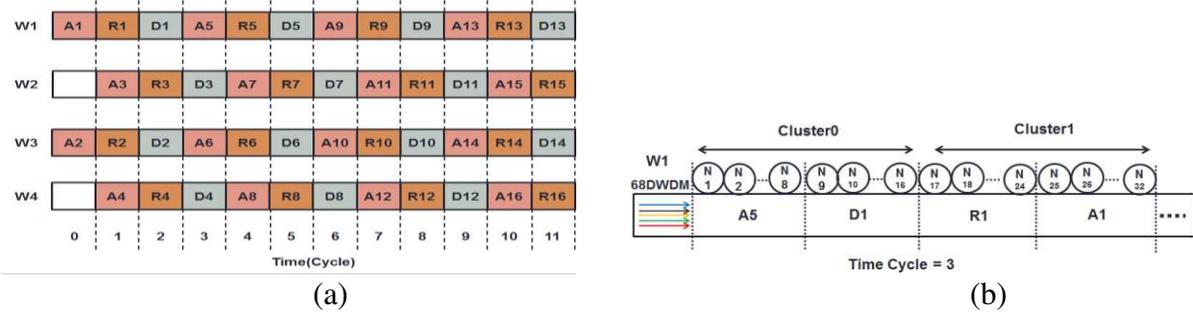


Figure 8 (a) Timing diagram of arbitration in UltraNoC, which shows distribution of arbitration (Ai), receiver prediction (Ri) and data slots (Di) across four MWMR waveguide groups (W1 – W4); (b) distribution of different slots within MWMR waveguide group W1 at time cycle 3.

For powering the waveguides, we use a broadband off-chip laser source with a laser power controller (LSWC). The LSWC has groups of ring modulators capable of injecting different wavelengths in different clock cycles. As we use 68 DWDM in the first MWMR waveguide and 64 DWDM in the remaining three MWMR waveguides of an MWMR waveguide group, there are 260 ring modulators in each ring modulator group in the LSWC. These ring modulators either allow (in non-resonance mode) or remove (in resonance mode) their corresponding wavelengths from the waveguide group. Therefore, these ring modulators in the LSWC inject either of the four arbitration wavelengths (i.e., λ_0 - λ_3) in the arbitration slot, the remaining 64 receiver selection wavelengths (i.e., λ_4 - λ_{67}) in the receiver selection slot, and the same 64 receiver selection wavelengths (i.e., λ_4 - λ_{67}) in the data slot (*SwiftNoC* uses the same set of wavelengths for the receiver selection process and data transfers). Further on-off switching time of a ring modulator is about 3.1 ps [9], which is less than one clock cycle (i.e. 400ps) at 2.5GHz frequency. The laser controller also has control logic that can alter the rate of injection of arbitration wavelengths into each waveguide group. The LSWC's ring modulators and its control logic are assumed to be fabricated on-chip [14]. More details about the LSWC are presented in the following subsections and overhead analysis is given in Section 2.4.1.

2.3.2. MWMR CONCURRENT TOKEN STREAM ARBITRATION AND RECEIVER SELECTION IN ULTRANOC

In UltraNoC, all of the cores on a chip are partitioned into four clusters ($C_0 - C_3$) and each cluster is assigned a dedicated arbitration wavelength ($\lambda_0 - \lambda_3$). Each MWMR waveguide group is divided into a fixed number of time slots, based on the time taken by light to traverse the waveguide on a die. Based on geometric calculations, each pass of the MWMR waveguide takes 4 cycles in our architecture at 2.5GHz clock frequency. Thus we divide each MWMR waveguide group into 8 time slots (4 time slots for each of first and second pass). The time slots are further classified into three types: *arbitration slot*, *receiver selection slot*, and *data slot*.

Figure 8(a) shows an example of the distribution of time slots across 4 MWMR waveguide groups (Note: a minimum of 8 MWMR waveguide groups are used in our architecture; we only show 4 in the figure for brevity). As per the explanation provided in the previous subsection, in the arbitration slot, the LSWC injects the arbitration wavelengths of clusters, selectively using a modulator group to dedicate the arbitration slot to a particular cluster. Further in UltraNoC each receiving node N_i is assigned a receiver selection wavelength λ_{i+4} (see section 2.3.1). Thus after a sending node grabs an arbitration wavelength in the arbitration slot, it gets access to the next receiver selection slot which initially has all the receiver selection wavelengths injected by the LSWC. In this receiver selection slot the sending node removes all the receiver selection wavelengths except the one corresponding to its receiving node using its modulators bank. Subsequently, in the next data slot, the sending node modulates data on the 64 wavelengths ($\lambda_4 - \lambda_{67}$) in each waveguide group assigned for data transfer. In the receiving portion of the MWMR waveguide (second pass of dual-coiled MWMR waveguide) whenever a receiver selection slot reaches a receiving node (N_i), the receiving node only switches-on its detector corresponding to

its receiver selection wavelength λ_{i+4} . Whenever a receiving node detects its receiver selection wavelength in the receiver selection slot, it switches-on its remaining detectors to receive data in the next data slot.

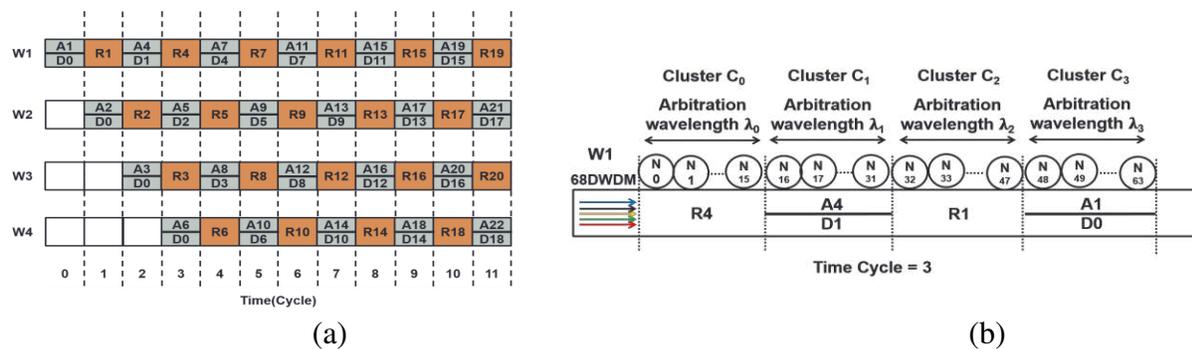


Figure 9 (a) Timing diagram of arbitration in *SwiftNoC*, which shows distribution of arbitration (A_i), receiver selection (R_i), and data slots (D_i) across four MWMM waveguide groups (W1 – W4); (b) distribution of different slots within MWMM waveguide group W1 at time cycle 3.

We illustrate this sending and receiving process with an example. In Figure 8(b) suppose N_1 in cluster C_0 needs to send data to N_{31} in cluster C_1 that has a corresponding receiver selection wavelength λ_{35} . N_1 first grabs arbitration wavelength λ_0 which is dedicated to cluster C_0 , in the arbitration slot. N_1 then modulates in the next receiver selection slot, such that only λ_{35} (the dedicated wavelength for receiver selection of N_{31}) is made available by removing all the wavelengths except λ_{35} (using its ring modulators) in that receiver selection slot. On the receiving end, all the detecting nodes which are in the receiver selection slot switch-on their detectors for the corresponding receiver selection wavelengths (e.g. Nodes N_{24} to N_{31} switch-on detectors with resonance wavelengths λ_{28} to λ_{35}). Thus at N_{31} , only the detector for wavelength λ_{35} is switched on in the receiver selection slot. Once λ_{35} is detected, N_{31} prepares to receive data in the next data slot by switching on the remaining detectors in that node.

Figure 8(b) shows a snapshot of the position of different slots in the MWMM waveguide group W1 at time cycle 3 for the example in Figure 8(a). As our architecture divides each pass of an MWMM waveguide group into 4 slots, each slot covers 16 nodes in a particular time instance. The stream of tokens (i.e., stream of arbitration slots with arbitration wavelengths dedicated to a specific cluster) on concurrent slots in waveguide groups allows multiple nodes to inject packets simultaneously on the same MWMM waveguide, resulting in extremely high channel utilization of each MWMM waveguide group. Further in our architecture, multiple nodes can inject packets across different MWMM waveguide groups as well, as each node has a separate modulator and detector bank on each MWMM waveguide group. As each arbitration slot covers 16 nodes on each MWMM waveguide at a time instant, therefore, two or more nodes (up to 16 nodes) from the same cluster can arbitrate for the same arbitration slot on each MWMM group. Ultimately one of them gets access to the arbitration slot by grabbing the arbitration wavelength. We employ a round-robin arbiter within each cluster to resolve this contention among the 16 nodes within a cluster, and avoid starvation.

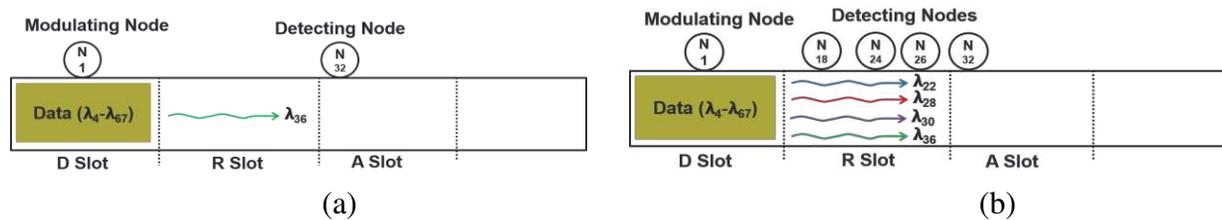


Figure 10 (a) Transmission of unicast data from Node N_1 to Node N_{32} in *SwiftNoC*, which shows receiver selection wavelength λ_{36} in receiver selection slot (R Slot) of the MWMM waveguide; (b) Multicast of data from Node N_1 to Nodes N_{18} , N_{24} , N_{26} , and N_{32} in *SwiftNoC*, which shows respective receiver selection wavelengths λ_{22} , λ_{28} , λ_{30} , and λ_{36} in receiver selection slot (R Slot) of the MWMM waveguide.

2.3.3. IMPROVED MWMMR CONCURRENT TOKEN STREAM ARBITRATION IN SWIFTNOC

As discussed in the previous subsection, UltraNoC uses separate wavelengths for arbitration (4 wavelengths $\lambda_0 - \lambda_3$) and data transfer (64 wavelengths $\lambda_4 - \lambda_{67}$). Further from Figure 8 it can be observed that arbitration slots (A_{i+1}) and data slots (D_i) in UltraNoC are adjacent to each other. We propose to overlap arbitration and data slots in our improved *SwiftNoC* architecture. This overlapping mechanism effectively reduces the number of slots for each data transfer from 3 in UltraNoC to 2 in the *SwiftNoC* architecture. Figure 9 illustrates the *SwiftNoC* version of the timing diagram for UltraNoC shown in Figure 8. Figure 9(a) shows an example of the distribution of time slots across 4 MWMMR waveguide groups, with overlapped arbitration and data slots. Further, Figure 9(b) shows the position of different slots in the MWMMR waveguide group W1 at time cycle 3 for the example in Figure 9(a) with arbitration and data slots overlapped. The *SwiftNoC* architecture improves utilization of MWMMR waveguides compared to the MWMMR waveguide utilization in UltraNoC, which results in an increase in available bandwidth and reduced average packet latency in comparison to the UltraNoC architecture.

2.3.4. MULTICASTING OF MESSAGES IN SWIFTNOC

In CMP's with cache coherency support, multicast traffic makes up a significant portion of total network traffic. For example, in the MOESI cache coherence protocol, when a shared block is invalidated, an invalidate message must be multicast to all sharers of that particular shared block. The UltraNoC architecture presented in the subsections 2.3.1 and 2.3.2 will translate these multicast messages into several unicast messages and send them to their respective destination nodes. These unicast messages cause network congestion and may reduce its performance [70].

In *SwiftNoC*, we avoid such repeated unicast messages by providing multicasting support in its MWMM waveguides. Unlike Corona [11] and Firefly [12] architectures where all multicast messages are broadcast and transmitted to all nodes in the network, *SwiftNoC* enables multicasting to specific nodes in the network. This is realized as follows: each sending node in *SwiftNoC*, after removal of the arbitration wavelength from the arbitration slot, releases multiple receiver selection wavelengths corresponding to multiple receiving nodes in the next receiver selection slot (in contrast, in UltraNoC, a sender node, after removal of the arbitration wavelength from the arbitration slot, releases the wavelength of a single receiving node in the next receiver selection slot). In the immediately following data slot, the sending node modulates data which needs to be multicast to different receivers. To enable photonic multicast of data in MWMM waveguides, we partially de-tune the ring detectors from their resonating wavelengths [71], such that a portion of the photonic energy continues on in the MWMM waveguide to be absorbed in subsequent ring detectors. Multicasting thus requires higher laser power compared to unicasting so as to maintain sufficient photonic signal intensity for detection in the worst case, i.e., for the detectors of the last receiving node which receives the multicast data. Laser power injected into the MWMM waveguide for multicasting operation in *SwiftNoC* does not change with the number of nodes that need to receive the multicast message. We designed the laser source for the worst-case power loss, which occurs when all the receiving nodes receive a multicast message from a sending node. We have considered this extra laser power overhead when presenting energy consumption results for the *SwiftNoC* architecture in our results section. In this chapter, we do not consider optimizing laser power through a laser power management scheme. However, it is possible to integrate previously proposed laser power management schemes [14] [39] with our work, as these works are orthogonal to our work.

Figure 10(a) and (b) illustrate the difference between transmission of unicast and multicast messages in our *SwiftNoC* architecture. Suppose N_1 in cluster C_0 needs to multicast data to N_{18} , N_{24} , N_{26} , and N_{32} whose corresponding receiver selection wavelengths are λ_{22} , λ_{28} , λ_{30} , and λ_{36} respectively. N_1 first grabs arbitration wavelength λ_0 which is dedicated to cluster C_0 , in the arbitration slot. N_1 then modulates in the next receiver selection slot, such that only λ_{22} , λ_{28} , λ_{30} , and λ_{36} are made available by removing all the wavelengths except λ_{22} , λ_{28} , λ_{30} , and λ_{36} (using its modulators) in that receiver selection slot. At the receiver end at N_{18} , N_{24} , N_{26} , and N_{32} , the detectors for wavelengths λ_{22} , λ_{28} , λ_{30} , and λ_{36} respectively are switched on when these nodes are in the receiver selection slot. At N_{18} , once λ_{22} is detected in the receiver selection slot, the node prepares to receive data in the next data slot by partially de-tuning the ring detectors from its resonating wavelengths in that node. The partial de-tuning of ring detectors of N_{18} will remove a portion of light available in the MWMR waveguide leaving the remaining portion of light for the other detectors to absorb. Similarly, on detection of λ_{28} , λ_{30} , and λ_{36} , nodes N_{24} , N_{26} , and N_{32} respectively prepare to receive data in the next data slot. Our *SwiftNoC* architecture does not differentiate between unicast and multicast transmissions, as it always employs partial detuning to receive both unicast and multicast messages. To further improve channel utilization in the *SwiftNoC* architecture we adapt the inter-cluster bandwidth transfer mechanism from UltraNoC, as described in in the next subsection.

2.3.5. INTER-CLUSTER BANDWIDTH EXCHANGE IN SWIFTNOC

SwiftNoC support inter-cluster bandwidth transfers to further improve channel utilization and overall performance. As an example, if cluster C_0 does not need to transfer data, it transfers its bandwidth to a subsequent cluster C_1 . Similarly, any cluster can transfer its unused bandwidth

to the subsequent clusters. Figure 11 presents an overview of the bandwidth transfer technique. The last node in each cluster is provisioned with an extra ring modulator that is capable of injecting the arbitration wavelength of the next cluster. Whenever a ring detector of the last node of a cluster detects its own arbitration wavelength, and if this node does not have data to transfer, this indicates a case where the cluster has not used its bandwidth.

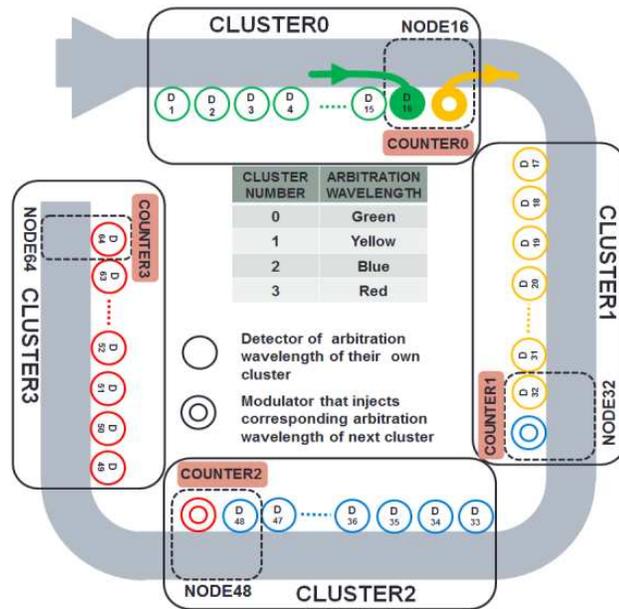


Figure 11 Bandwidth transfer technique: a cluster can transfer its unused bandwidth to the next cluster by absorbing its own arbitration wavelength and releasing arbitration wavelength of next cluster.

Figure 11 illustrates an example of the bandwidth transfer process. Clusters $C_0 - C_3$ are assigned with arbitration wavelengths highlighted with green, yellow, blue, and red respectively. The last node in the first three clusters (N_{15} , N_{31} , and N_{47}) is shown with an extra ring modulator that facilitates the injection of the arbitration wavelength of the next cluster. For this example, nodes in C_0 do not need to transfer any data in the current cycle. Then N_{15} , which is the last node in C_0 removes its clusters' arbitration wavelength (green) from the arbitration slot and injects the

arbitration wavelength of C_1 (yellow) so that nodes in C_1 can use this arbitration slot for sending data in the next available data slot. The bandwidth exchange mechanism performs arbitration wavelength conversion in an arbitration slot in one cycle, thus it has minimal delay/control overhead. The presence of additional microrings for bandwidth transfer mechanism does lead to more through losses on the MWMR waveguide which ultimately increases total laser power dissipation. This increase in laser power is included in the laser power dissipation of the 68 DWDM MWMR waveguide ($P_{\text{MWMR-MCT}}$) in Table 3. Figure 11 also shows counters at nodes N_{15} , N_{31} , N_{47} , and N_{63} that are used to count the number of arbitration wavelength conversions over a time interval. The next subsection presents more details about the need for these counters.

2.3.6. CLUSTER PRIORITY ADAPTATION WITH LSWC RECONFIGURATION

SwiftNoC also supports runtime alteration of allocated bandwidth to each cluster, to closely track changing application bandwidth needs, by altering the number of arbitration slots dedicated to each cluster (i.e., cluster priority). This is essential because while our bandwidth transfer technique can transfer unused arbitration slots from one cluster to another in the direction of the concurrent arbitration token stream flow (e.g., C_0 to C_1), it lacks the ability to transfer bandwidth in the opposite direction (e.g., C_1 to C_0). To overcome this limitation, we design a cluster priority adaptation mechanism to more comprehensively manage cluster bandwidth allocations over time. This mechanism also helps to minimize laser power by intelligently reducing the total number of injected arbitration slots for runtime scenarios with low bandwidth (traffic) requirements. The cluster priority adaptation technique consists of 3 main steps, they are:

Step 1: Determination of wavelength conversion count: Each cluster $C_0 - C_3$ has associated weights $W_0 - W_3$, which determine the proportion of arbitration slots (and consequently bandwidth or

priority) assigned to the cluster. Initially, these weights can be set to be equal, i.e., 0.25 each. At runtime, whenever the last node in a cluster performs a wavelength conversion from its current cluster arbitration wavelength to the next cluster arbitration wavelength, a counter (shown in Figure 11) is incremented. This conversion event represents the case where an unused arbitration slot (bandwidth) is transferred from one cluster to another. Over a time interval T, the recorded wavelength conversion counts $WCC_0 - WCC_3$ from each cluster are then used to determine the unused bandwidth for each cluster.

Step 2: Calculation of excess arbitration slots: The wavelength conversion count values of different clusters show the aggregate number of excess arbitration slots, which includes excess arbitration slots of the present cluster along with the excess arbitration slots of predecessor clusters. The excess arbitration slots for the i^{th} cluster (ES_i) are calculated using Eq. (1) shown below, by subtracting the cluster wavelength conversion count of the predecessor cluster (WCC_{i-1}) from the wavelength conversion count of the cluster under consideration (WCC_i). ES_i values can also be negative, when a cluster consumes a greater number of arbitration slots (made available by predecessor clusters) than its allocated arbitration slots. Such a cluster has a deficit of arbitration slots.

$$ES_i = \begin{cases} WCC_i, & i = 0 \\ WCC_i - WCC_{i-1}, & i > 0 \end{cases} \quad (1)$$

Step 3: Setting new weight (priority) for each cluster: Based on the estimation of excesses and deficits in arbitration slots assigned across clusters, this final step attempts to adjust weight values of each cluster to eliminate the excesses and deficits. To determine the new weight of the i^{th} cluster $W_{i(\text{next})}$ for the upcoming time interval, we must subtract the excess weight EW_i of the cluster from its current weight $W_{i(\text{current})}$. We can calculate EW_i by dividing the excess arbitration slots

of the i^{th} cluster (ES_i), calculated in Eq. (1), by the total number of arbitration slots released in the time interval T , which we denote as K . The equations below show these calculations:

$$EW_i = ES_i/K \quad (2)$$

$$W_i(next) = W_i(current) - EW_i \quad (3)$$

Based on the values of the new weights, the LSWC changes the distribution of arbitration wavelengths injected for the next time interval T , such that a cluster with a higher weight will receive more arbitration wavelengths, presenting it with more opportunities to use the waveguides for data transfer. These weight values are communicated to all clusters, so that arbiters can adjust their local counters to match the new arbitration slot profile in the waveguides.

2.4. EXPERIMENTS

2.4.1. EXPERIMENTAL SETUP

To evaluate our proposed *SwiftNoC* architecture, we compared it to a traditional electrical mesh (EMesh) based NoC as well as to four state-of-the-art photonic crossbar NoCs: UltraNoC with concurrent token stream arbitration [26], Flexishare with token stream arbitration [13], Firefly with reservation-assisted single-write-multiple-reader (R-SWMR) data waveguides [12], and Corona with an enhanced token-slot arbitration [67]. We modeled and simulated the architectures at a cycle-accurate granularity with a SystemC-based NoC simulator, for two CMP platform complexities: 64-core and 256-core. We used random synthetic traffic for preliminary analysis of the proposed architectures. Subsequently, we used the PARSEC benchmark suite [43] to create multi-application workloads, with clusters running parallelized versions of different benchmarks from this suite, for more detailed comparisons.

Table 2 Memory intensity classification of PARSEC benchmarks

Application	Representation	Workload Type
Blackscholes	BS	Compute intensive
Bodytrack	BT	Compute intensive
Vips	VI	Compute intensive
Dedup	DU	Compute intensive
Freqmine	FQ	Hybrid
Ferret	FR	Hybrid
Fluidanimate	FA	Hybrid
X264	X264	Hybrid
Streamclusters	SC	Memory intensive
Canneal	CA	Memory intensive
Facesim	FS	Memory intensive
Swaptions	SW	Memory intensive

Table 2 shows the PARSEC benchmarks we considered, classified into three categories according to their memory intensities. Compute intensive benchmarks spend most of the time computing and less time communicating with memory; whereas memory intensive applications spend a larger portion of their execution time communicating with memory and less time computing within cores. Hybrid intensity benchmarks demonstrate both compute and memory intensive phases. We created 12 multi-application workloads from these benchmarks. Each workload combines 4 benchmarks, and the memory intensity of the workloads varies across the spectrum, from compute intensive to memory intensive. As an example, the SC-BT-BS-VI workload combines parallelized implementations of Streamclusters (SC), Bodytrack (BT), Blackscholes (BS), and Vips (VI), and executes them in clusters C_0 , C_1 , C_2 , and C_3 respectively. Each parallelized benchmark is executed on a group of 16 cores and 64 cores, in the 64-core and 256-core CMP platforms, respectively. A system-level simulation was performed with the open-source GEM5 [72] architectural simulator with 64 and 256 ARM-based cores running parallelized PARSEC benchmarks, to generate traces that were fed into our cycle-accurate NoC simulator. More details about cache sizes, cache associativity, cache coherence, issue policy, memory

controllers, and DRAM sizes which we have considered to generate these traces are presented in Table 1. We set a “warm-up” period of 100-million instructions and then captured traces for the subsequent 1-billion instructions. Then a trace-driven simulation was performed with our cycle-accurate SystemC based NoC simulator.

Table 3 Energy and losses for photonic devices [73], [74]

Energy consumption type	Energy
$E_{dynamic}$	0.42 pJ/bit
$E_{logic-dyn}$	0.18 pJ/bit
Static power per waveguide group	Power
P_{MWMR} (with 64 DWDM)	3.73 W
$P_{MWMR-MCT}$ (with 68 DWDM)	5.32 W
$P_{MWMR-MCT}$ (with 64 DWDM)	4.95 W
P_{MWSR} (with 64 DWDM)	2.35 W
P_{SWMR} (with 64 DWDM)	1.15 W
Photonic loss type	Loss (in dB)
Microring through	0.02
Waveguide propagation per cm	1
Waveguide coupler/splitter	0.5
Chip coupling	1
Waveguide bending loss	0.005 per 90^0

Table 4 Photonic hardware comparison

Architecture	Waveguides	Ring Modulators	Ring Detectors	PNoC Area (in mm²)
SwiftNoC-8	32	131,640	131,584	24.50
SwiftNoC-16	64	263,280	263,168	49.01
SwiftNoC-32	128	526,560	526,336	98.01
UltraNoC-8	32	131,640	131,584	24.51
UltraNoC-16	64	263,280	263,168	49.01
UltraNoC-32	128	526,560	526,336	98.01
FLEXISHARE	33	131,080	131,648	24.58
FIREFLY	64	4,096	28,672	10.25
CORONA	257	1,032,256	20,416	113.47

Table 5 Properties of various PNoC Architectures

Architecture	Waveguide Type	Arbitration Scheme	Multicast Ability	Packet Size
SwiftNoC-8	MWMR	Improved Concurrent Token Stream	Yes	512 bits
SwiftNoC-16	MWMR	Improved Concurrent Token Stream	Yes	512 bits
SwiftNoC-32	MWMR	Improved Concurrent Token Stream	Yes	512 bits
UltraNoC-8	MWMR	Concurrent Token Stream	No	512 bits
UltraNoC-16	MWMR	Concurrent Token Stream	No	512 bits
UltraNoC-32	MWMR	Concurrent Token Stream	No	512 bits
FLEXISHARE	MWMR	2-Pass Token Stream	No	512 bits
FIREFLY	SWMR	-	Yes	512 bits
CORONA	MWSR	Fair Token Slot	No	512 bits

We targeted 32nm and 22nm process technologies for the 64-core and 256-core CMPs, respectively. Based on the geometric calculation of the waveguides for a 20mm×20mm chip dimension, we estimated the time needed for light to travel from the first to the last node in a single pass of the MWMR waveguide group in *SwiftNoC* as 4 cycles at 2.5 GHz clock frequency. The same clock and 4 cycle round trip time is also applicable to the waveguides in the UltraNoC, Flexishare, Firefly, and Corona photonic crossbar NoCs. Throughout our analysis we use a flit size of 64 bits for EMesh and a total packet size of 512 bits. Further we also consider a similar packet size of 512 bits for all photonic NoC architectures. We consider data modulation at both clock edges to enable simultaneous transfer of 512 bits in a single cycle, in the *SwiftNoC*, UltraNoC, Flexishare, Firefly, and Corona architectures. We presented architectural information about all the PNoC architectures used in our analysis in Table 5.

The static and dynamic energy consumption of electrical routers is based on results obtained from the open-source DSENT tool [75]. Energy consumption of various photonic components for all the photonic NoC architectures are adapted from photonic device characterizations in line with state-of-the-art proposals [30] [73] [74] and shown in Table 3. Here $E_{dynamic}$ is the energy/bit for

modulators and photodetectors and $E_{\text{logic-dyn}}$ is the energy/bit for the driver circuits of modulators and photodetectors. P_{MWMR} , P_{MWSR} , and P_{SWMR} are the static power consumption of an MWMR, MWSR, and SWMR waveguide group, respectively, which includes the power overhead of ring resonator thermal tuning. The static power consumption of each MWMR waveguide group with multicasting enabled in the *SwiftNoC* architecture is shown as $P_{\text{MWMR-MCT}}$. Further we have considered a power dissipation overhead of 0.12W and 0.1W in the electrical circuits of the 68 and 64 DWDM MWMR-MCT waveguides respectively, to realize partial detuning while still maintaining acceptable bit-error-rate (BER) as low as 10^{-9} , based on the estimation from the prior work [71]. We consider a ring heating power of $15 \mu\text{W}$ per ring and detector responsivity of 0.8 A/W [73].

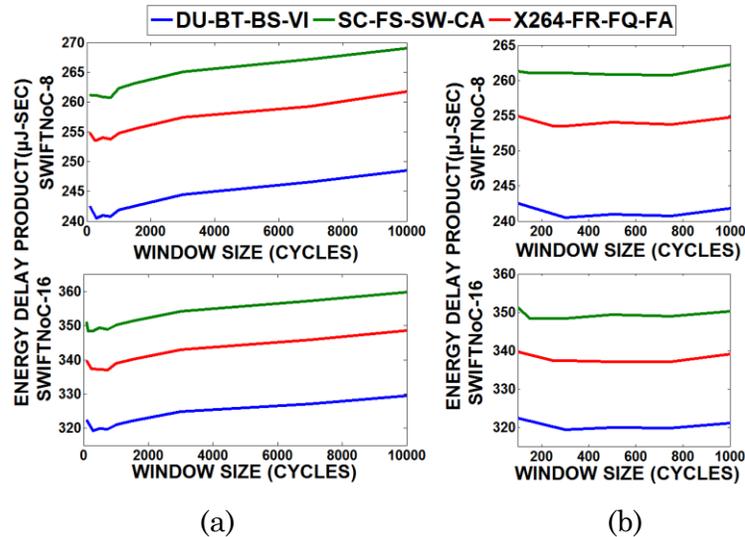


Figure 12 Energy-delay-product (EDP) comparison for SwiftNoC-8 and SwiftNoC-16 in a 64-core CMP with time interval window sizes (a) 100-10000 cycles (b) 100-1000 cycles (zoomed version of Figure 12(a)).

To compute laser power consumption, we calculated photonic loss in components, which sets the photonic laser power budget and correspondingly the electrical laser power. Lastly, based

on our gate-level analysis, area and power overheads are estimated to be 0.011mm^2 and 0.023W respectively for the electrical circuitry (e.g., adders, multipliers, comparators) in the LSWC for our priority adaptation mechanism for *SwiftNoC* at 32nm. We set the reconfiguration delay overhead in *SwiftNoC* to be 20 cycles to account for the time to transfer wavelength conversion counter values from each cluster to the LSWC, time to determine new priority weights of each cluster, and time to update these values in the arbiters in each cluster.

2.4.2. EXPERIMENTAL RESULTS

2.4.2.1. SENSITIVITY ANALYSIS TO DETERMINE OPTIMAL RECONFIGURATION WINDOW SIZE

Our first set of experiments presents a sensitivity analysis to explore the optimal dynamic bandwidth and priority reconfiguration time interval window size in *SwiftNoC*. We explore two variants of our architecture: *SwiftNoC-8* which uses 8 waveguide groups and *SwiftNoC-16* which uses 16 waveguide groups. Figure 12(a) shows the energy-delay-product (EDP) for three multi-application PARSEC workloads in *SwiftNoC-8* and *SwiftNoC-16*, with window lengths varying from 100 to 10000 cycles. In this analysis to compute EDP we have considered energy consumption of PNoC only (core + cache energy consumption is not considered in our analysis). The three workloads were chosen to possess high, medium, and low aggregate memory intensity, to explore the impact of varying memory intensities on window size. At a particular window size, this figure shows higher EDP for memory intensive workloads compared to compute intensive workloads, as memory intensive workloads route more packets in *SwiftNoC*, which increases their dynamic energy consumption and average packet latency (due to increased network congestion), thereby increasing overall EDP. Also, for both memory and compute intensive workloads, a large

window size should intuitively result in lower reconfiguration overhead but will also result in less reactivity to changing application traffic demands which ultimately increases average packet latency and EDP as well; while a small window size will result in higher reconfiguration overhead with higher energy consumption in the reconfiguration hardware and increased EDP, but better adaptivity to changing application traffic. A careful observation of the plot in Figure 12 shows that for compute intensive workloads (i.e., DU-BT-BS-VI) EDP is lower for a larger window size, whereas for memory intensive workloads (i.e. SC-FS-SW-CA) EDP is lower at smaller window sizes. However, there is an overlap region from 300 cycles to 750 cycles that can be observed from Figure 12(b) (which is the zoomed version of Figure 12(a) between window sizes 100-1000 cycles) where EDP is low for both memory and compute intensive workloads. Additionally, results for average throughput and latency also indicate worsening performance beyond 750-1000 cycles. Thus, we set reconfiguration time interval window size in *SwiftNoC* to 300 cycles, to balance reconfiguration overhead and performance. Our analysis of the reconfiguration time interval window size for UltraNoC also indicates an optimal EDP at around 300 cycles, thus we also set a 300-cycle reconfiguration time interval window size for UltraNoC.

2.4.2.2. RESULTS OF 64-CORE SYSTEM FOR SYNTHETIC TRAFFIC

Our second set of experiments targets a 64-core CMP platform with a synthetic benchmark that utilizes a uniform random traffic pattern. In uniform random traffic, cores arbitrarily generate packets to random destination cores in the CMP. Cache coherency or multicast traffic is not considered in this analysis with random traffic. We compare network throughput, average packet latency, and EDP of *SwiftNoC* with the electrical mesh (EMesh), UltraNoC with concurrent token arbitration [26], Flexishare with token stream arbitration [13], Firefly with reservation-assisted

single write multiple reader (R-SWMMR) data waveguides [12], and Corona with token-slot arbitration [67]. Later in this section we also present comparison results of *SwiftNoC* architecture (*SwiftNoC-MCT*) for various percentages of multicast traffic to the total traffic of the network.

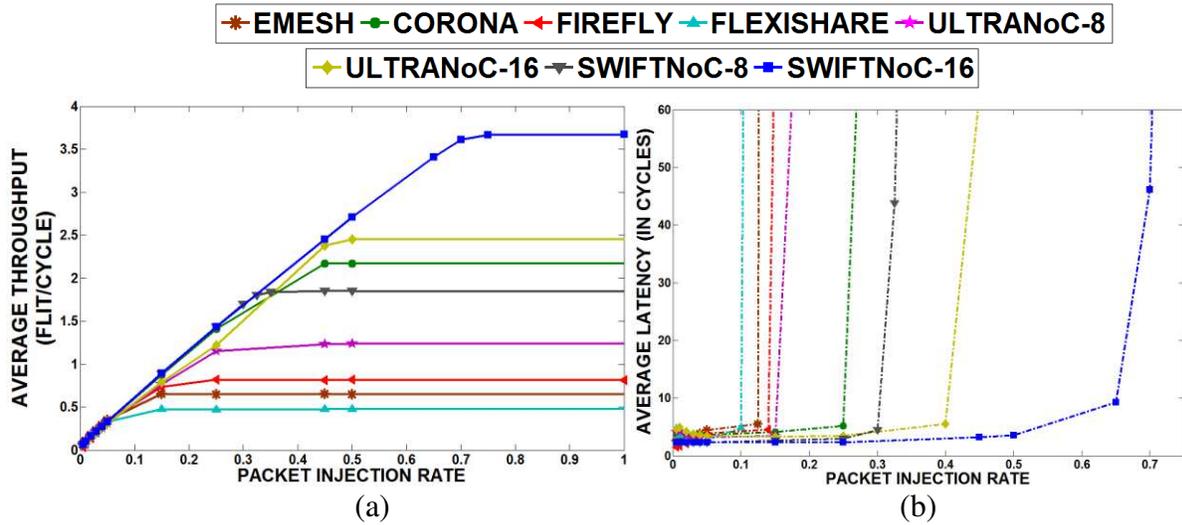


Figure 13 (a) Average throughput, (b) average latency comparison of *SwiftNoC-8* and *SwiftNoC-16* with UltraNoC-8, UltraNoC-16, Flexishare, Firefly, Corona, and EMesh architectures for a 64-core CMP. Results are shown for uniform random traffic.

The average throughput for uniform random traffic in the 64-core CMP is shown in Figure 13(a). It can be observed that *SwiftNoC* with 8 MWMR waveguides (*SwiftNoC-8*) has 4.2 \times and 1.7 \times higher throughput compared to Flexishare and UltraNoC-8 with the same number of MWMR data waveguides. Even though Flexishare uses MWMR waveguides and time division multiplexing (TDM) as in *SwiftNoC*, there are significant differences between these architectures. In Flexishare, arbitration wavelengths corresponding to each MWMR data waveguide are injected serially into an arbitration waveguide. A node that grabs a token in the arbitration waveguide gets exclusive access to the corresponding MWMR data waveguide which leads to underutilization of the MWMR waveguide. In contrast, *SwiftNoC-8* uses improved concurrent token stream

arbitration with TDM, such that each MWMMR waveguide with multiple arbitration, receiver selection, and data slots can be accessed concurrently by multiple nodes to facilitate simultaneous transfer of multiple packets to improve MWMMR waveguide utilization. Moreover, unlike *SwiftNoC*, Flexishare does not support the priority reconfiguration and bandwidth exchange mechanisms. Further, compared to UltraNoC, *SwiftNoC* uses an improved version of concurrent token stream arbitration which increases the data rate in MWMMR waveguides by overlapping arbitration and data slots. This overlapping mechanism effectively reduces the number of slots for each data transfer from 3 in UltraNoC to 2 in the *SwiftNoC* architecture and increases throughput for *SwiftNoC-8* compared to UltraNoC-8. The throughput of *SwiftNoC-8* is 2.8× higher than the throughput of EMesh, as our architecture uses faster silicon photonic waveguides for data communication compared to slower electrical links. *SwiftNoC-8* also has 2.2× higher throughput compared to Firefly. This is because *SwiftNoC-8* for a 64-core CMP is an all optical MWMMR crossbar, which transfers all of its data at near light speed whereas Firefly is a hybrid photonic network, where a significant portion of data traverses through slower electrical links. The *SwiftNoC* configurations with 16 waveguide groups (*SwiftNoC-16*) with approximately twice the number of microring resonators in *SwiftNoC-8* provides even better throughput than the UltraNoC-16 (with 16 MWMMR waveguide groups), Firefly, Flexishare, and EMesh architectures. *SwiftNoC-16* has 1.6×, 8.4×, 4.5×, and 5.6× greater throughput compared to UltraNoC-16, Flexishare, Firefly, and EMesh respectively for the 64-core CMP.

Figure 13(a) also shows that Corona has greater throughput compared to *SwiftNoC-8*. This is because Corona has 64 MWSR waveguides and an MWMMR arbitration waveguide to facilitate communication between 64 cores, which utilizes approximately four times the number of microring resonators compared to *SwiftNoC-8* (as shown in

Table 4 earlier). However, *SwiftNoC* with 16 waveguide groups (*SwiftNoC-16*) and approximately half the number of microring resonators of Corona has 1.9× better throughput compared to Corona. *SwiftNoC-16* uses MWMMR waveguides with improved concurrent token stream arbitration and achieves higher data rates through simultaneous data transfers using TDM, which is not possible with the MWSR waveguides used in Corona. The enhanced token-slot arbitration [67] in Corona requires a fixed time gap between two arbitration slots to set up data for transmission, which reduces available time slots to send data.

From Figure 13(b) it can be seen that in terms of average latency, *SwiftNoC-8* and *SwiftNoC-16* have better performance compared to Flexishare and Firefly. Flexishare with its inefficient arbitration scheme has underutilized MWMMR waveguides which increases overall packet latency. In contrast, *SwiftNoC* with an improved concurrent token stream arbitration mechanism, bandwidth transfer mechanism, and cluster priority adaptation mechanism increases MWMMR waveguide utilization and reduces wait time for packets, which in turn reduces latency. Further, in the reservation assisted Firefly architecture, a sender needs extra cycles to broadcast reservation flits to all the destination nodes, so that destination node can tune in on the corresponding SWMMR data waveguides to receive the data in the following cycles. These extra cycles lead to lower data rates in SWMMR waveguides of the Firefly architecture and increase its average latency compared to *SwiftNoC*. The improved concurrent token stream arbitration scheme used in *SwiftNoC* enables higher communication parallelism compared to token based arbitration in Corona, which explains the lower latency in *SwiftNoC* compared to Corona. Lastly, *SwiftNoC* has lower latency compared to UltraNoC with the same number of waveguides, as the improved concurrent token stream arbitration in *SwiftNoC* increases the data rate in its MWMMR waveguides by overlapping arbitration and slots, which helps reduce average latency.

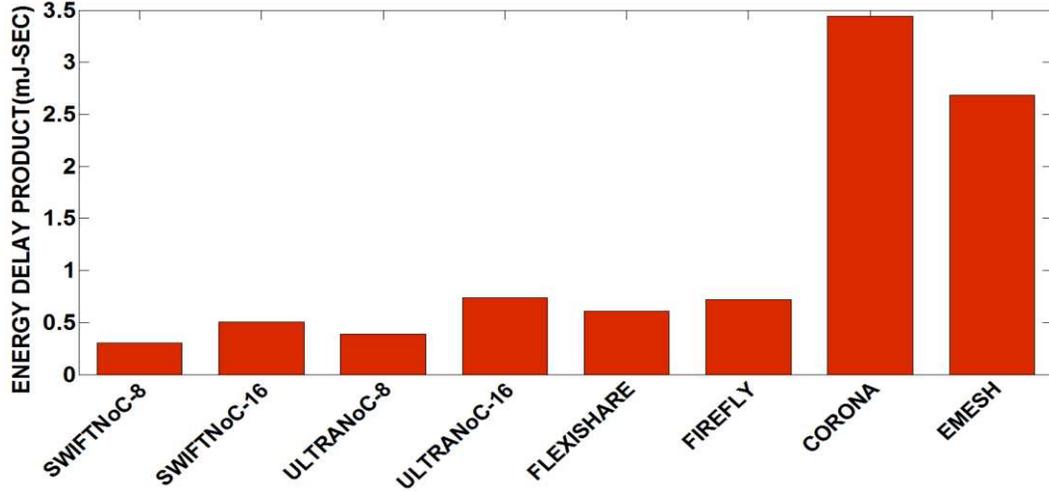


Figure 14 Energy-delay-product (EDP) comparison of *SwiftNoC-8* and *SwiftNoC-16* with UltraNoC-8, UltraNoC-16, Flexishare, Firefly, Corona, and EMesh architectures for a 64-core CMP. Results are shown for uniform random traffic with packet injection rate of 0.7.

Figure 14 summarizes the EDP of our *SwiftNoC* architectures with UltraNoC-8, UltraNoC-16, Flexishare, Firefly, Corona, and EMesh for the uniform synthetic traffic pattern. These results are generated for a packet injection rate of 0.7, for which the throughputs for all of these compared architectures are saturated. Energy consumption includes static, dynamic and laser energy for every architecture. From these results it can be seen that *SwiftNoC-8* has 49%, 57%, 91%, and 88%, and *SwiftNoC-16* has 17%, 30%, 85%, and 81% lower EDP compared to Flexishare, Firefly, Corona, and EMesh, respectively. Corona has more EDP compared to *SwiftNoC-8* and *SwiftNoC-16* as it uses more number of microring resonators as shown in

Table 4, which in turn leads to more static energy consumption. The lower EDP of *SwiftNoC-8* and *SwiftNoC-16* compared to Firefly is due to higher energy consumption in the electrical network of the Firefly architecture. Although *SwiftNoC-8* has similar number of microring resonators compared to Flexishare, the improvements in average latency for *SwiftNoC-8* due to improved sharing contribute to its lower EDP. *SwiftNoC-8* also has 21% and 58% lower EDP compared to UltraNoC-8 and UltraNoC-16 respectively and *SwiftNoC-16* has 31% lower EDP

compared to UltraNoC-16. Despite higher dynamic power due to an increase in its data rate (increase in number of data modulation and detection events), *SwiftNoC* has higher EDP savings compared to UltraNoC with similar number of MWMM waveguides because of overlapped arbitration and data slots which reduce average packet latency and EDP. *SwiftNoC-8* also has half the number of microring resonators compared to UltraNoC-16, which in turn results in higher relative static energy consumption and EDP for UltraNoC-16.

As a final set of experiment with synthetic traffic we present average throughput, average packet latency, and EDP for the *SwiftNoC* architecture for various percentages of multicast traffic to total traffic in the network. The average throughput for uniform random traffic for a 64-core CMP with *SwiftNoC* having 16 MWMM waveguides is shown in Figure 15(a) for 10% (*SwiftNoC-16-MCT-10*), 20% (*SwiftNoC-16-MCT-20*), 30% (*SwiftNoC-16-MCT-30*), 40% (*SwiftNoC-16-MCT-40*), and 50% (*SwiftNoC-16-MCT-50*) of multicast traffic out of the total traffic in the network. It can be observed that with increase in multicast traffic percentage there is a monotonic increase in throughput for *SwiftNoC*. From Figure 15(b), for average latency, *SwiftNoC-16-MCT-50* has lower latency compared to *SwiftNoC-16-MCT-40*, *SwiftNoC-16-MCT-30*, *SwiftNoC-16-MCT-20* and *SwiftNoC-16-MCT-10*. With the increase in multicast traffic in the network, *SwiftNoC* enables sharing of photonic signals during multicast with partial de-tuning of microring resonators, which allows multiplexing of data streams. This in turn increases throughput and reduces latency with simultaneous delivery of packets to their respective destinations, and shows the adaptability of the proposed *SwiftNoC* architecture for higher multicast traffic rates.

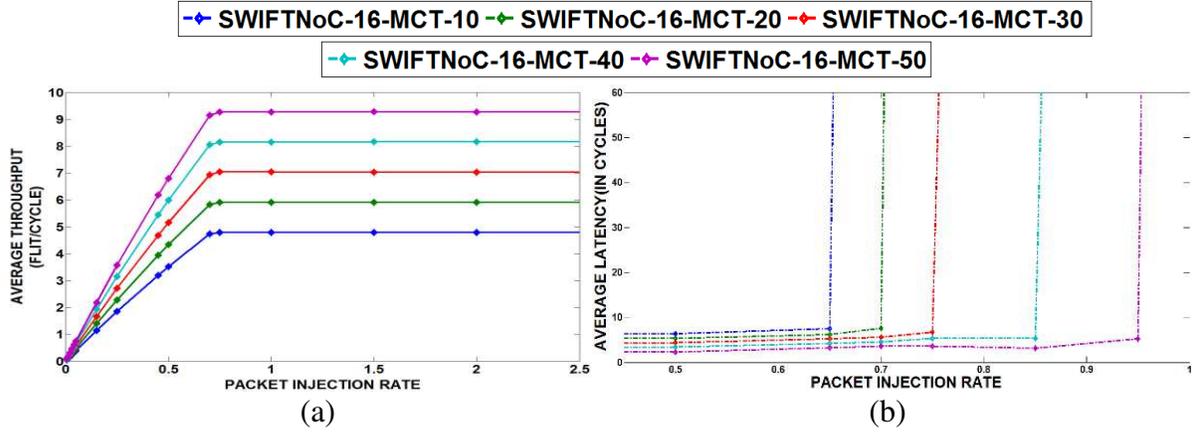


Figure 15 (a) Average throughput (b) average latency comparison of *SwiftNoC-16* with random multicast traffic having 10% (*SWIFTNoC-MCT-10*), 20% (*SWIFTNoC-MCT-20*), 30% (*SWIFTNoC-MCT-30*), 40% (*SWIFTNoC-MCT-40*), and 50% (*SWIFTNoC-MCT-50*) of multicast messages for a 64-core CMP.

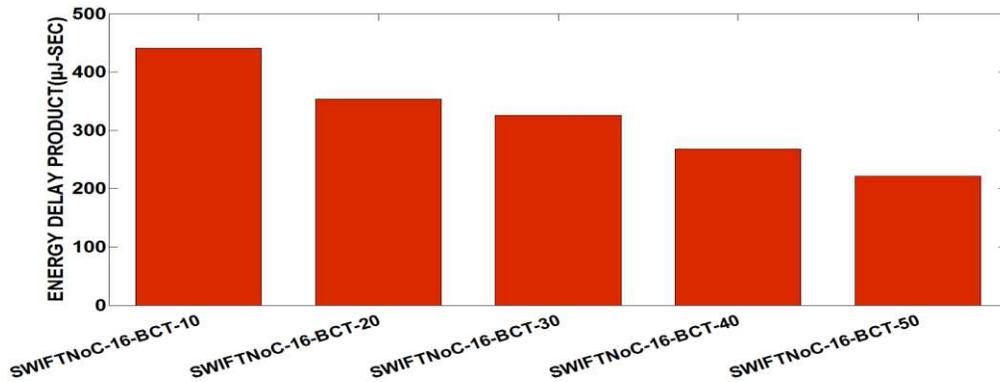


Figure 16 Energy-delay-product (EDP) comparison of *SwiftNoC-16-MCT-10*, *SwiftNoC-16-MCT-20*, *SwiftNoC-16-MCT-30*, *SwiftNoC-16-MCT-40*, and *SwiftNoC-16-MCT-50* for a 64-core CMP. Results are shown for uniform random traffic with different percentages of multicast traffic at packet injection rate of 0.95.

Figure 16 shows the EDP comparison between *SwiftNoC-16-MCT-10*, *SwiftNoC-16-MCT-20*, *SwiftNoC-16-MCT-30*, *SwiftNoC-16-MCT-40*, and *SwiftNoC-16-MCT-50* for a 64-core CMP for a uniform synthetic traffic pattern. These results are generated for a packet injection rate of 0.95, where the throughput for all of these compared architectures is saturated. From these results it can be seen that *SwiftNoC-16-MCT-50* has 50%, 38%, 32%, and 18% lower EDP compared to *SwiftNoC-16-MCT-10*, *SwiftNoC-16-MCT-20*, *SwiftNoC-16-MCT-30*, and *SwiftNoC-16-MCT-40*.

Although *SwiftNoC-16-MCT-50* has similar number of microring resonators as *SwiftNoC-16-MCT-10*, *SwiftNoC-16-MCT-20*, *SwiftNoC-16-MCT-30*, and *SwiftNoC-16-MCT-40*, but with higher multicast traffic in the network, *SwiftNoC-16-MCT-50* performs more number of multicasts through partial de-tuning of microring resonators and significantly increases the delivery rate of packets, which reduces overall average latency and decreases EDP.

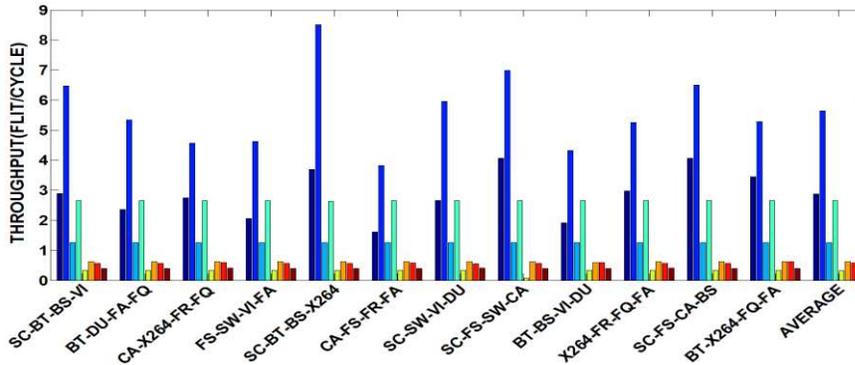
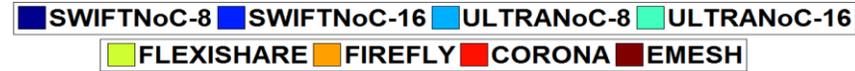
2.4.2.3. EXPERIMENTAL ANALYSIS WITH 64-CORE CMP

Our next set of experiments target a 64-core CMP platform and compare network throughput, average packet latency, and energy-per-bit (EPB) of *SwiftNoC* with the electrical mesh (EMesh), UltraNoC, Flexishare, Firefly, and Corona architectures.

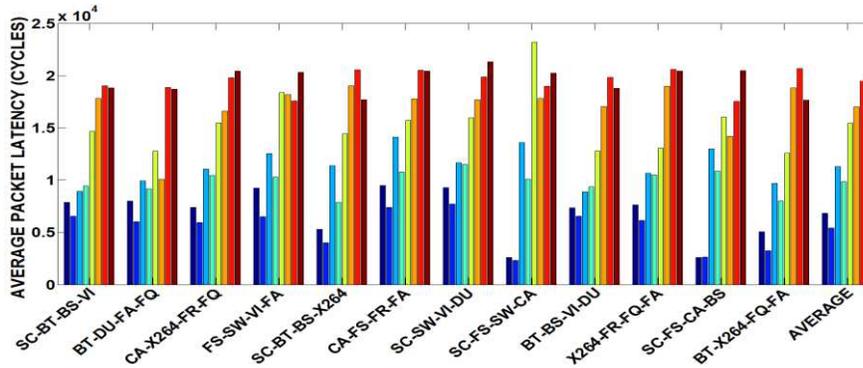
Figure 17(a)-(c) show the results of this study, with all results normalized with respect to the EMesh results. From the throughput comparison in Figure 17(a), it can be observed that, not surprisingly, all photonic NoCs provide better throughput than EMesh, due to the presence of higher bandwidth photonic links. Further, *SwiftNoC* when compared to EMesh, UltraNoC, Flexishare, Firefly, and Corona has even better throughput improvements for PARSEC benchmark traffic than with synthetic traffic. From Figure 17(a) it can be seen that *SwiftNoC-8* has $7.8\times$ greater throughput compared to EMesh, as well as $9.1\times$ and $2.3\times$ greater throughput compared to Flexishare and UltraNoC-8, with the same number of MWMR waveguides. In Flexishare, as explained in the previous subsection token stream arbitration hinders utilization of MWMR waveguides; whereas in *SwiftNoC*, multiple arbitration, reservation and data slots are available concurrently in an MWMR waveguide, such that each MWMR waveguide can be accessed simultaneously by multiple nodes. The improved concurrent token stream arbitration which reduces the number of time slots for each data transfer and multicasting which enables

simultaneous transfer of multiple messages contribute to increase in the throughput of *SwiftNoC-8* compared to *UltraNoC-8*. *SwiftNoC-8* also provides $5.1\times$ higher throughput than *Corona*. This is because of instances in *Corona*, where multiple sender nodes attempt to communicate with a single receiver node (e.g., memory controller). Such instances result in the sender nodes attempting to access the single MWSR waveguide connected to the receiver, creating a significant imbalance among MWSR waveguides, with the other waveguides being underutilized while packets get queued waiting for the waveguide connected to the receiver. *SwiftNoC* avoids such an imbalance with its use of more efficient MWMR waveguides and improved arbitration. *SwiftNoC-8* also provides $4.6\times$ more throughput than *Firefly*. *SwiftNoC* for a 64-core CMP is an all optical MWMR crossbar, which transfers data entirely over photonic links and thus has increased throughput compared to *Firefly* which is a hybrid photonic network, where a significant portion of data traverses through slower electrical links. The bandwidth transfer mechanism and cluster priority adaption mechanism in *SwiftNoC* also increase available bandwidth and contribute to increase in throughput.

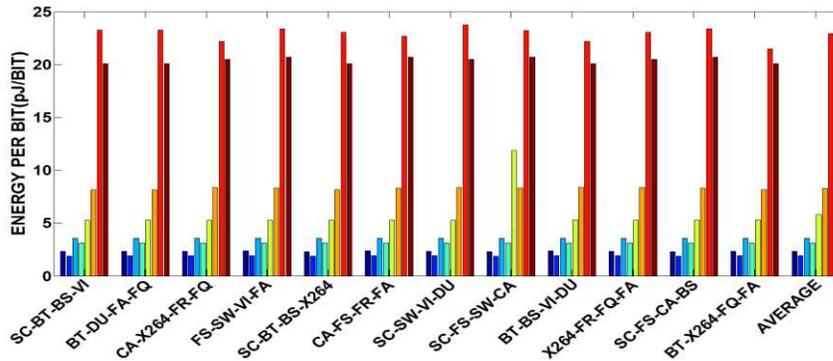
SwiftNoC with 16 MWMR waveguide groups (*SwiftNoC-16*) with approximately twice the number of microring resonators of *SwiftNoC-8*, provides even better throughput than the other architectures. *SwiftNoC-16* has $2.2\times$, $17.8\times$, $9.1\times$, $9.9\times$, and $14.2\times$ higher throughput compared to *UltraNoC-16*, *Flexishare*, *Firefly*, *Corona* and *EMesh*, respectively. The improvement is somewhat higher for memory intensive workloads than for compute intensive workloads. The large throughput improvement for *SwiftNoC* is a direct consequence of improved concurrent token stream arbitration, multicasting, avoiding unused bandwidth by transferring it to cores that need it the most, and using the bandwidth transfer and priority alteration mechanisms at runtime.



(a)



(b)



(c)

Figure 17 (a) Average throughput (b) average packet latency (c) average energy-per-bit (EPB) comparison of *SwiftNoC-8* and *SwiftNoC-16* with other architectures for a 64-core CMP. Results are shown for multi-application PARSEC workloads.

These mechanisms also improve the average packet latency in *SwiftNoC* as shown in Figure 17(b), by reducing the time spent waiting for access to the photonic waveguides. On average *SwiftNoC-8* has 39.8%, 55.7%, 59.7%, 65.1%, and 65.3% lower average packet delay over UltraNoC-8, Flexishare, Firefly, Corona and EMesh, respectively for the different multi-application workloads. On the other hand, *SwiftNoC-16* has 45.1%, 63.9%, 68.4%, 72.1%, and 72.4% lower average packet delay over UltraNoC-16, Flexishare, Firefly, Corona and EMesh, respectively. From these results, we can surmise that average latency improvements of *SwiftNoC* over UltraNoC, Flexishare, Firefly, Corona and EMesh with benchmark traffic and synthetic traffic follow similar trends.

Figure 17(c) shows the EPB comparison between the architectures. It can be observed that on average *SwiftNoC-8* has 34%, 25%, 59%, 72%, 90%, and 89%, and *SwiftNoC-16* has 47%, 38%, 67%, 77%, 92%, and 91%, lower EPB compared to UltraNoC-8, UltraNoC-16, Flexishare, Firefly, Corona, and EMesh respectively. Most of the energy in the photonic architectures was consumed in the form of static energy. From

Table 4 presented earlier, it can be observed that *SwiftNoC-8* has 75% lesser number of MRs, whereas *SwiftNoC-16* has 50% lesser number of MRs compared to Corona. This allows *SwiftNoC-8* and *SwiftNoC-16* to have lower EPB compared to Corona. Further *SwiftNoC-8* with similar number of MRs as Flexishare maintains lower EPB with more efficient utilization of its MWMR waveguides through its concurrent token stream arbitration. On the other hand, despite *SwiftNoC-16* using more hardware than Flexishare, it has lower EPB compared to Flexishare because of more efficient arbitration and multicasting, the bandwidth transfer mechanism, and priority alteration mechanism. Firefly has higher EPB even though it uses lesser number of microring resonators compared to *SwiftNoC-8* and *SwiftNoC-16*. Firefly being a hybrid network, consumes most of its

energy in the electrical network and this in turn increases overall EPB compared to the *SwiftNoC* architectures. All the variants of our *SwiftNoC* architecture have lower EPB compared to EMesh, as our architectures use energy efficient photonic links for data transfer instead of power hungry electrical links. Although *SwiftNoC-8* and *SwiftNoC-16* use power hungry multicast MWMR waveguides (see Table 3), the increase in data rate due to efficient multicasting and improved concurrent token stream arbitration decreases EPB of these architectures compared to different variations of the UltraNoC architecture.

2.4.2.4. SCALABILITY ANALYSIS WITH 256-CORE CMP

Our final set of experiments explores the scalability of *SwiftNoC*. We considered a larger 256-core CMP platform by increasing the core concentration in each tile to four to enable higher traffic injection into the network for this scalability study. We evaluated NoC throughput, average packet latency, and EPB for all photonic NoCs. In addition to *SwiftNoC-8* and *SwiftNoC-16*, we also considered a *SwiftNoC-32* variant of our architecture with 32 MWMR waveguide groups. Similarly, we also considered an additional UltraNoC-32 variant of the UltraNoC.

Figure 18(a)-(c) show the results of these experiments. From the throughput results in Figure 18(a) it can be seen that on average *SwiftNoC-8* has 2.4 \times , 1.2 \times , 7.3 \times , 4.1 \times , 7.2 \times , and 4.8 \times , *SwiftNoC-16* has 5.1 \times , 2.5 \times , 16.5 \times , 9.1 \times , 16.3 \times and 10.8 \times , and *SwiftNoC-32* has 8 \times , 3.7 \times , 25.4 \times , 14.3 \times , 25.1 \times , and 16.6 \times , greater throughput compared to UltraNoC-8, UltraNoC-16, Flexishare, Firefly, Corona and EMesh. Further *SwiftNoC-16* and *SwiftNoC-32* have 1.3 \times and 2.1 \times higher throughput compared to UltraNoC-32, respectively. The improvements in throughput for *SwiftNoC* over UltraNoC, Firefly, Flexishare, Corona, and EMesh are even better for the 256-core CMP, than in the 64-core CMP case. With the increase in core count, the amount of traffic injected into the network increases and *SwiftNoC* with its better utilized MWMR waveguides effectively

handles this traffic whereas UltraNoC, Firefly, Flexishare, Corona, and EMesh end up moving to the saturation region (throughput will not increase), which explains throughput improvements for *SwiftNoC*.

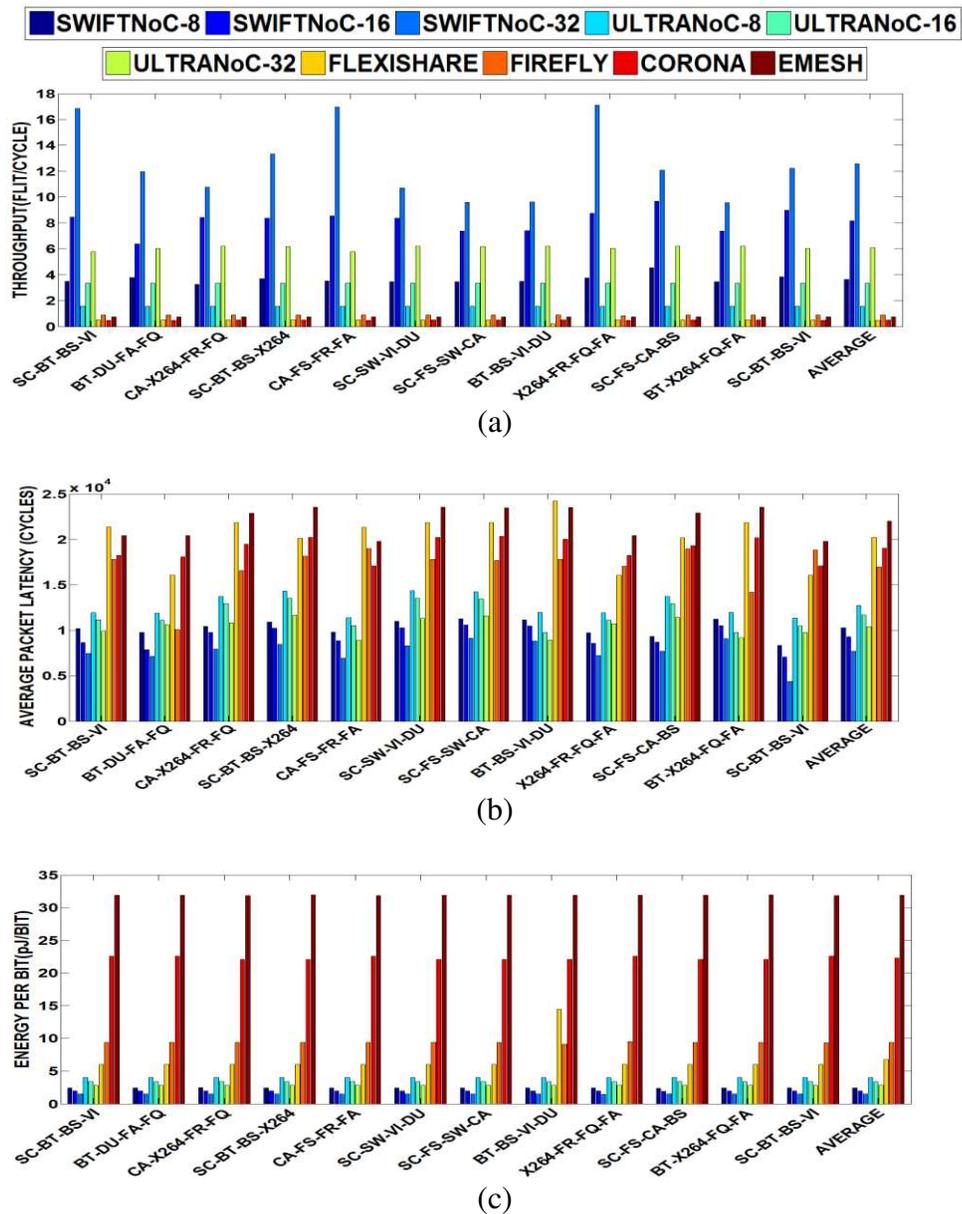


Figure 18 (a) Average throughput (b) average packet latency (c) average EPB comparison of *SwiftNoC-8*, *SwiftNoC-16*, and *SwiftNoC-32* with other architectures for a 256-core CMP. Results are shown for multi-application PARSEC workloads.

From Figure 18(b), it can be seen that on average *SwiftNoC-8* has 19%, 12%, 7%, 49%, 39%, 46%, and 53%, *SwiftNoC-16* has 27%, 20%, 11%, 54%, 45%, 51% and 57%, and *SwiftNoC-32* has 39%, 34%, 26%, 62%, 54%, 59% and 64% lower latency compared to UltraNoC-8, UltraNoC-16, UltraNoC-32, Flexishare, Firefly, Corona, and EMesh architectures respectively. Lastly, Figure 18(c) shows that on average *SwiftNoC-8* has 39%, 28%, 14%, 63%, 74%, 89% and 92%, *SwiftNoC-16* has 51%, 42%, 30%, 71%, 79%, 91%, and 94%, and *SwiftNoC-32* has 62%, 55%, 46%, 77%, 84%, 93%, and 95% lower EPB compared to UltraNoC-8, UltraNoC-16, UltraNoC-32, Flexishare, Firefly, Corona, and EMesh architectures respectively. The average packet latency and EPB improvements in *SwiftNoC* are higher for the 256-core CMP compared to the 64-core CMP. The greater volume of traffic in the 256-core system increases packet wait time in the sending nodes across all the architectures. *SwiftNoC* is able to reduce this wait time with its efficient arbitration, multicasting, bandwidth transfer, and priority alteration mechanisms, to achieve better average latency improvements over the UltraNoC, Flexishare, Firefly, Corona, and EMesh architectures. Despite the increase in energy consumption for all the architectures when going from the 64-core CMP to the 256-core CMP, the EPB of the *SwiftNoC* architecture has greater improvements over UltraNoC, Flexishare, Firefly, Corona, and EMesh architectures because of its higher packet delivery rate.

2.4.2.5. SUMMARY OF RESULTS AND OBSERVATIONS

From the results presented in the previous sections, we can summarize that our proposed *SwiftNoC* architecture can achieve better performance with less hardware compared to existing state-of-the-art photonic NoCs, for CMP platforms with low as well as high core counts. *SwiftNoC* achieves higher performance and energy-efficiency by efficiently utilizing MWMR waveguides with its

improved arbitration scheme, bandwidth transfer mechanism, and cluster priority adaptation. *SwiftNoC* also improves upon UltraNoC due to its more aggressively concurrent token stream arbitration scheme which increases utilization of MWMR waveguides compared to UltraNoC. The *SwiftNoC* architecture's ability to efficiently multicast messages across different nodes in the network using its multicast-friendly MWMR waveguides also contribute to its higher performance with lower energy consumption.

2.5. CONCLUSIONS

In this chapter, we presented the *SwiftNoC* photonic NoC architecture which is an improved version of the UltraNoC architecture, with more efficient channel sharing among cores with an aggressive concurrent token stream-based arbitration strategy and more efficient multicast support. *SwiftNoC* supports the ability to dynamically transfer bandwidth between clusters of cores and to re-prioritize multiple co-running applications to further improve channel utilization and adapt to time-varying application performance goals. *SwiftNoC* improves throughput by up to 25.4× while reducing latency by up to 72.4% and EPB by up to 95% over state-of-the-art solutions. *SwiftNoC* also scales well with increasing core counts on a chip.

3. BIGNOC: ACCELERATING BIG DATA COMPUTING WITH APPLICATION-SPECIFIC PHOTONIC NETWORK-ON-CHIP ARCHITECTURES

In the era of big data, high performance data analytics applications are frequently executed on large-scale cluster architectures to accomplish massive data-parallel computations. Often, these applications involve iterative machine learning algorithms to extract information and make predictions from large data sets. Multicast data dissemination is one of the major performance bottlenecks for such data analytics applications in cluster computing, as terabytes of data need to be distributed frequently from a single data source to hundreds of computing nodes. To overcome this bottleneck for big data applications, we propose *BiGNoC*, a manycore chip platform with a novel application-specific photonic network-on-chip (PNoC) fabric. *BiGNoC* is designed for big data computing and exploits multicasting in photonic waveguides. For high performance data analytics applications, *BiGNoC* improves throughput by up to 9.9× while reducing latency by up to 88% and energy-per-bit by up to 98% over two state-of-the-art PNoC architectures as well as a broadcast-optimized electrical mesh NoC architecture, and a traditional electrical mesh NoC architecture.

3.1. BACKGROUND, MOTIVATION, AND CONTRIBUTION

Large-scale data analytics applications represent some of the most data-intensive workloads in the emerging domain of big data computing. Most of the high-performance data analytics applications e.g., cancer genome analysis, stock market predictions, consumer product recommendations, disaster forecasting, etc. involve iterative execution of various machine learning algorithms. These iterative machine learning algorithms for large-scale data analytics tasks often

run on a MapReduce framework [76] implemented either in the cloud or on commodity clusters in datacenters.

Recently, Hadoop [77] and Spark [78] based distributed frameworks are being increasingly used for MapReduce implementations on cloud services. However, wide-spread security exploits and higher off-loading time with cloud computing have driven several organizations to build their own datacenters for big data processing [77]- [78]. Such datacenters are safer from intrusion with lower off-loading time, but according to the Hamilton's cost model [79] the overheads due to power dissipation, power distribution, and cooling in such datacenters with commodity processors can be quite significant. A specialized manycore processor solution in which a large number of cores are interconnected through an efficient on-chip network can reduce such overheads and lead to improved system performance, compared to commodity processors. *This motivates us to design a customized chip manycore processor (CMP) platform to more efficiently run the iterative machine learning algorithms for big data processing.*

The iterative algorithms in big data processing with MapReduce execute on multiple master and servant cores and take thousands of iterations to produce the desired output. Each iteration typically consists of three phases [80] (Figure 19). In the initial multicast phase (Figure 19(a)) a master node (MN), which consists of one or more master cores, multicasts a large feature set of model parameters to one or more servant nodes (SN; each with one or more servant cores) that perform computations based on the parameters. While computing, these servant nodes may need to exchange or shuffle data with other servant nodes. This phase is called the shuffle phase (Figure 19(b)). Lastly, in the aggregation phase (Figure 19(c)), all the servant nodes update and send their partial results to the master node. The master node aggregates this partial data to produce the multicasting data for the next iteration.

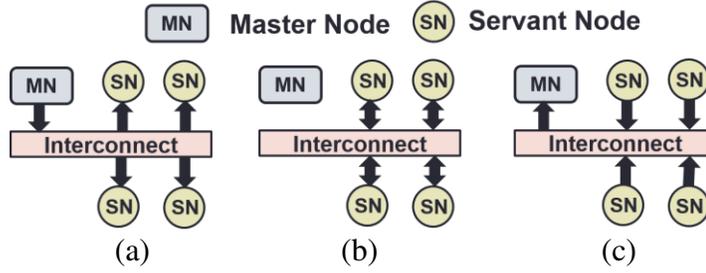


Figure 19 MapReduce (a) multicast phase, (b) shuffle phase, and (c) aggregation phase of communication while executing iterative machine learning algorithms for large-scale data analytics applications.

Multicasting is a performance bottleneck in executing large scale data analytics applications that have large fan-out, big data sizes, and take a large number of iterations to achieve convergence. For example, the K-nearest neighbor algorithm for breast cancer prediction and prognosis [81] requires multicasting of approximately 200 MB of sampled cancer genomic features in each iteration, from 100 image samples, each of size 2MB. As the typical number of iterations is more than 1000, the total multicasting data is in the order of hundreds of gigabytes. Another example is the alternating least squares algorithm for Netflix movie rating prediction, which involves 385MB of data being distributed to servant nodes per iteration, over hundreds of iterations [82]. This computation thus involves tens of gigabytes of multicast data. *These examples motivate the need for supporting efficient multicasting for big data workload execution scenarios.*

Recent developments in the fabrication of CMOS-compatible on-chip photonic interconnects have opened up the possibility of redesigning emerging manycore processing architectures, especially for big data applications. On-chip photonic interconnects provide several prolific advantages over their conventional metallic counterparts, including the ability to communicate at near light speed, larger bandwidth density by using dense wavelength division multiplexing (DWDM), and lower power dissipation [71]. These advantages motivate us to consider using photonic links for inter-core communication in CMPs that run the iterative

algorithms for big data processing. Further, a few prior works [11], [12], [13], [71] have emphasized the importance of multicasting in photonic waveguides to improve data communication rates, and proposed photonic network-on-chip (PNoC) architectures that enable inter-core communication with multicast-enabled waveguides. The multicasting capability of photonic interconnects further inspires us to use them in CMPs optimized for big data processing.

In this chapter, we present a novel application-specific PNoC architecture for manycore chips, called *BiGNoC*, to execute large-scale data analytics applications with high throughput and ultra-low latency. To the best of our knowledge, this is the first work that attempts to design PNoCs to tackle iterative machine learning algorithm based large-scale data analytics applications in CMPs. Our novel contributions are:

- We devise a master-servant cluster based communication fabric (*MSNoC*) with dedicated channels for master-to-servant and servant-to-master communication;
- We design a hierarchical manycore *BiGNoC* architecture with multiple *MSNoCs* to execute any combination of high performance large-scale data analytics applications;
- We evaluate *BiGNoC* by comparing it with two previously proposed PNoCs, as well as a broadcast optimized electrical mesh NoC, and a traditional electrical mesh NoC for multiple real-world big data applications [83], [84], [85], [86].

3.2. RELATED WORK

Photonic interconnects utilize several photonic devices such as microring resonators (MRs) as modulators, detectors, and switches; photonic waveguides; splitters, and trans-impedance amplifiers (TIAs). Each MR has a unique resonance wavelength in the utilized DWDM spectrum in a waveguide (typically consisting of 64 or less wavelengths) that it can couple to and work

correctly with. This resonant nature of an MR allows it to be used as a filter or a switch. A filter MR is used to filter and drop its resonance wavelength on to a photodetector, whereas a switch MR is used to route the propagation of a resonant wavelength signal between two waveguides.

Several PNoC architectures have been proposed to date (e.g., [11]- [13], [61], [62], [87]) that use on-chip photonic interconnects with MR modulators to modulate electrical signals at the source node on to photonic signals, which then travel through a photonic waveguide, and arrive at MR detectors at the destination node where the photonic signals are detected and electrical signals recovered. Several efforts have explored high throughput crossbar PNoCs that provide non-blocking connectivity, e.g., [11]- [13], [62] using different types of photonic waveguides such as Multiple-Write-Single-Read (MWSR), Single-Write-Multiple-Read (SWMR), and Multiple-Write-Multiple-Read (MWMR). A few works exploit multicasting in SWMR [12] and MWSR [11] waveguides to improve the performance of PNoC architectures with cache coherence traffic (e.g., in the MOESI coherence protocol, when a shared block is invalidated, an invalidate message must be multicast to all sharers). *However, no prior work has attempted to design PNoCs to optimize iterative machine learning algorithm-based large-scale data analytics applications in CMPs.*

Several architectures have been explored recently to address large-scale data analytics applications. A PENC manycore architecture consisting of 192 small processing cores was proposed in [88], which can work as a co-processor in tandem with a general-purpose CPU to accelerate big data processing. A low-power manycore architecture for a modern big-data stream mining applications is proposed in [89] that is able to cope with the dynamic nature of the input data stream while consuming limited power. A parallel CMP architecture called SpiNNaker based on a customized electrical NoC to implement spiking neural networks was proposed in [90]. The

cores in this architecture are connected by a modified version of the torus topology, whereas the inter-chip topology is a 2D triangular mesh with 6-port routers. A neural network architecture called EMBRACE is proposed in [91] which integrates a 2D array of interconnected neural tiles surrounded by I/O blocks and adopts a hierarchical mesh-based topology to connect neural tiles. Furthermore, it uses a region-based routing scheme in each network layer to direct messages to destination nodes. Some works have demonstrated reconfigurable neural networks on a broadcast-aware mesh NoC architecture [92], [93]. A theoretical analysis for determining a preferred interconnect architecture for general purpose configurable emulation of spiking neural networks is presented in [92] and shows that mesh NoC using multicast is the most suitable architecture for a wide range of neural network topologies. A cluster-based reconfigurable NoC architecture for neural networks is presented in [93], which employs a reconfigurable communication fabric that efficiently handles multicast communication. In [94], a CPU-GPU architecture was presented with an electrical ring network to better execute large-scale data analytics applications, but this ring interconnect is known to be inefficient for large-scale systems. A hybrid (wired+wireless) on-chip interconnect based CPU-GPU architecture was proposed in [95] for large-scale data analytics applications. The authors in [96] propose Melia, which is an FPGA-based MapReduce architecture. *None of the abovementioned prior works explore the impact of using photonic interconnects for big data processing as part of the on-chip network. Our goal in this chapter is to show, for the first time, how PNoC architectures can be designed and customized for manycore chips, to meet the unique communication requirements of big data analytics applications.*

3.3. MASTER-SERVANT CLUSTER ARCHITECTURE

High-performance data analytics applications use a set of iterative machine learning algorithms for data predictions. A machine learning job may take hundreds or thousands of

iterations to converge to a solution. On a CMP, each iteration starts with the multicast of a big data set of model parameters from a master core to all the servant cores. Then the servant cores sometimes exchange data among themselves while processing their received data, thus creating inter-servant traffic. Lastly, each servant updates the model parameters partially and sends these model parameters to the master node. These partial results are aggregated at the master node to form the global model parameters for the computations in the subsequent iteration. Thus, execution of large-scale data-intensive applications requires dedicated hardware with master cores, servant cores, and an interconnection fabric between the masters and servants. In this section, we describe the architecture of a new master-servant cluster based communication fabric (*MSNoC*), in which master cores are connected to servant cores via photonic communication channels.

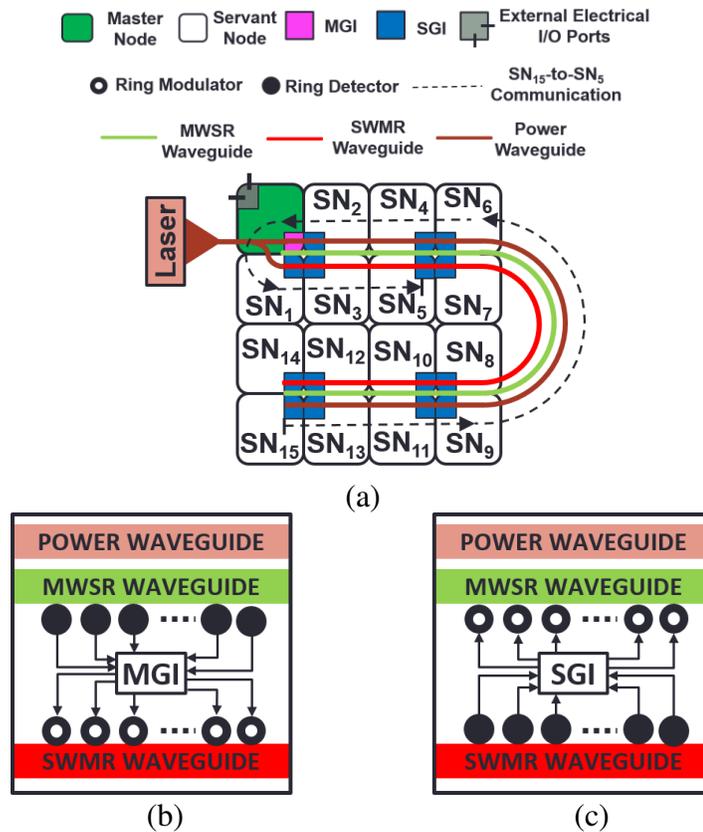


Figure 20 (a) MSNoC layout with SWMR, MWSR, and power waveguides (b) master gateway interface (MGI) (c) servant gateway interface (SGI).

In our *MSNoC* architecture, a node (N) is defined as an entity consisting of four cores. A node can either be a master node (MN; with four master cores) or a servant node (SN; with four servant cores). Each master core in an MN has a private L1 and L2 cache, whereas each servant core in an SN has only a private L1 cache. Every MN and SN is attached to a gateway interface (GI) module that facilitates transfers between the core-cache layer and the interconnection network layer. A detailed layout of the *MSNoC* is shown in Figure 20(a), where 16 nodes are arranged in a 4×4 grid. Among these 16 nodes, a single node is an MN and the remaining nodes are SNs (i.e., SN₁ to SN₁₅). The master GI (MGI) and servant GI (SGI) are shown in Figure 20(b) and (c), respectively, and discussed further in Sections 3.3.1-3.3.3. Communication between cores within a node (MN or SN) uses a 5×5 on-chip electrical router, where four of its input and output (I/O) ports are connected to four cores (master or servant) and the fifth I/O port is connected to the GI module associated with the node. A round-robin arbitration scheme is used within each node for communication between cores and the GI.

Communication between SNs and MNs is accomplished using SWMR and MWSR waveguides (Sections 3.3.1-3.3.3). There is also a power waveguide that runs in parallel with the SWMR and MWSR waveguides. This power waveguide carries all the wavelengths used for data traversal in the waveguides. A 1×2 splitter is used to split power from the power waveguide to SWMR waveguides as shown in Figure 34(a). In addition, a series of 1×2 splitters along the power waveguide are used to supply power to the modulators that are used to write data on to the MWSR waveguides. The splitting losses due to these splitters are considered in the laser power calculations of *MSNoC* (see Section 3.6). Our *MSNoC* with a group of 16 nodes (with 64 cores) has dedicated access to main memory via a memory controller at the MN. This is similar to the processor used in Sunway TaihuLight [97], which has dedicated main memory access for every 64 cores. The

micro-architectural parameters of nodes and cores in an *MSNoC* cluster are summarized in Table 6. In the following three subsections, we present more details about the interconnects that are used to enable communication between the MNs and SNs of an *MSNoC*.

Table 6 Micro-Architectural Parameters for MSNoC Cluster

Number of nodes per cluster	16 (1 MN and 15 SNs)
Number of cores	64 (4 per node)
Servant Core:	
L1 I-Cache size/ Associativity	16KB/Direct Mapped Cache
L1 D-Cache size/ Associativity	16KB/Direct Mapped Cache
Master Core:	
L1 I-Cache size/ Associativity	32KB/Direct Mapped Cache
L1 D-Cache size/ Associativity	32KB/Direct Mapped Cache
L2 Cache size/ Associativity	128KB/ Direct Mapped Cache
L2 Coherence	MOESI
Frequency	5 GHz
Issue Policy	In-order
Memory controllers	1
Main memory	8GB; DDR5@30ns

3.3.1. MN-to-SN COMMUNICATION IN MSNOC CLUSTER

As discussed earlier, the interconnection network between the master and servant cores plays a crucial role towards achieving faster execution of large-scale data analytics applications on an *MSNoC* cluster. As the communication from master cores to servant cores has significant periods of multicast traffic, this motivates us to use multicast enabled photonic waveguides in our *MSNoC* cluster, to enable faster master-servant communication. As shown in Figure 20(a), in an *MSNoC* cluster we use a multicast enabled Single-Write-Multiple-Read (SWMR) waveguide group to enable communication from a single MN to multiple SNs, where each waveguide group has four SWMR waveguides. The SWMR waveguide group in an *MSNoC* starts from an MN and passes through all of the SNs (i.e., SN₁-SN₁₅) in the cluster Figure 20(a)) to enable MN-to-SN communication. An MN has the ability to write on the SWMR waveguide group using its ring modulators (see Figure 20(b), which shows modulators of an MN on SWMR waveguide), and all

the SNs are capable of reading from the SWMR waveguide group using their ring detectors (see Figure 20(c), which shows detectors of an SN on SWMR waveguide). To power these SWMR waveguides, we use a broadband off-chip laser source and a 1×4 splitter to split the laser power across the four SWMR waveguides. We also use 64 DWDM wavelengths in each of the four SWMR waveguides of the SWMR waveguide group. Therefore, in an SWMR waveguide group there are 256 modulators and 256 detectors in each MN and SN, respectively.

As all SNs are capable of receiving (reading) from an SWMR waveguide group during MN-to-SN communication, there is a need for receiver selection between SNs to ensure that only the designated receiver will receive data from the shared waveguide group. For receiver selection, each SWMR waveguide group is divided into a fixed number of time slots, based on the time taken by light to traverse the length of the waveguide on a die. Based on the geometric calculations considering a 100mm² chip area for a 64 core CMP at 22nm technology node, traversal of light through an SWMR waveguide group takes 2 cycles (i.e., 0.4 ns) in an *MSN_oC* cluster at 5GHz clock frequency. Therefore, we divide the SWMR waveguide group into 2 time slots, and each time slot is spread across 8 nodes (the node can either be an MN or SN), as shown in Figure 21. These time slots are further classified into two types: reservation cycle slots (RCS), and data cycle slots (DCS).

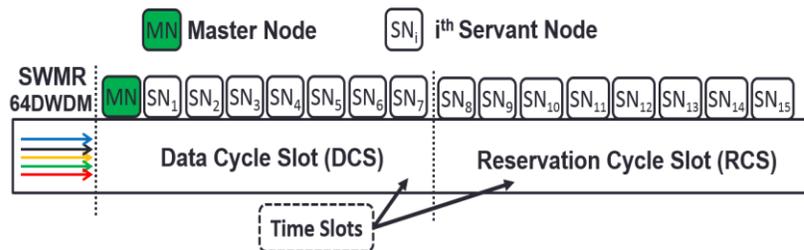


Figure 21 Distribution of reservation cycle and data cycle slots within SWMR waveguide to enable MN-to-SN communication.

In our reservation assisted MN-to-SN communication process, MNs send data to SNs in two cycles (Figure 21). In the reservation cycle, the MN reserves the SWMR waveguide group for an SN. Once the reservation is done, the MN sends data to the selected SN in the next cycle (i.e., data cycle). To perform the reservation, the MN uses the first SWMR waveguide in the SWMR waveguide group (this waveguide is shown in Figure 21). The remaining three SWMR waveguides in the SWMR waveguide group are used only in the data cycle to transfer data. Each SN_i is assigned a receiver selection wavelength λ_i , that is available in the first SWMR waveguide of the SWMR waveguide group. When an MN wants to send data to an SN, it gets access to the next RCS, which initially has all of the receiver selection wavelengths from the power waveguide. In this RCS, the MN uses its modulator bank to remove all of the receiver selection wavelengths except the one corresponding to the SN of interest. Subsequently, in the next DCS, the MN modulates data on the 256 wavelengths in four SWMR waveguides (as each SWMR waveguide uses 64 DWDM wavelengths ($\lambda_j - \lambda_{j+64}$)) of each SWMR waveguide group assigned for data transfer. *Therefore, our receiver selection mechanism prudently reuses the same set of wavelengths in the first SWMR waveguide of an SWMR waveguide group for reservation and data transmission.* On the receiving side of the SWMR waveguide group, whenever an RCS reaches an SN_i , it only switches on the detector which corresponds to its receiver selection wavelength λ_i located on the first SWMR waveguide of the SWMR waveguide group. Whenever an SN_i detects its receiver selection wavelength in the RCS, it switches on its remaining detectors not only on the first SWMR waveguide but also on the remaining three SWMR waveguides of the SWMR waveguide group to receive data in the next DCS.

We illustrate this sending and receiving process with a simple example. In Figure 22(a), suppose an MN needs to send data to SN_8 that has a corresponding receiver selection wavelength

λ_8 . The MN modulates in the next RCS, such that only λ_8 (the dedicated wavelength for receiver selection of SN_8) is made available by removing all of the wavelengths except λ_8 (using its modulators) in the first SWMR waveguide of the SWMR waveguide group. On the receiving end, all of the SNs which are in the RCS switch-on their detectors for the corresponding receiver selection wavelengths (e.g., nodes SN_8 to SN_{15} switch-on detectors with resonance wavelengths λ_8 to λ_{15} , respectively) in the first SWMR waveguide of the SWMR waveguide group. Therefore, at SN_8 only the detector for wavelength λ_8 is switched on in the RCS. Once λ_8 is detected, SN_8 prepares to receive data in the next DCS by switching on the remaining detectors not only on the first SWMR waveguide but also on the remaining three SWMR waveguides in the SWMR waveguide group in that node.

The receiver selection mechanism presented above can only transmit unicast messages, but while executing big data applications the MN will send not only unicast messages to a single SN but also multicast messages to multiple SNs. One possible solution is to translate these multicast messages into several unicast messages and send them to their respective SNs. But this can cause network congestion and reduce network performance [70]. Therefore, for MN to multiple SN communication in an *MSNoC*, we avoid such repeated unicast messages by providing multicasting support in the *MSNoC*'s SWMR waveguides.

Unlike Corona [11] and Firefly [12] PNoCs, where all multicast messages are broadcast and transmitted to all nodes in the network, *MSNoC* enables multicasting to specific nodes in the network. This is realized as follows: the MN in an *MSNoC* releases multiple receiver selection wavelengths into the first SWMR waveguide of the SWMR waveguide group (see Figure 22(b)) corresponding to multiple SNs in the next RCS. In the immediately following DCS, the MN modulates the data which needs to be multicast to different SNs on to four SWMR waveguides

within the SWMR waveguide group. To enable photonic multicast of data in SWMR waveguides, we partially de-tune the ring detectors from their resonating wavelengths [71], such that a portion of the photonic energy in the SWMR waveguide group continues to be absorbed in subsequent ring detectors. Multicasting thus requires higher laser power compared to unicasting so as to maintain sufficient photonic signal intensity for detection in the worst case, i.e., for the detectors of the last receiving node which receives the multicast data.

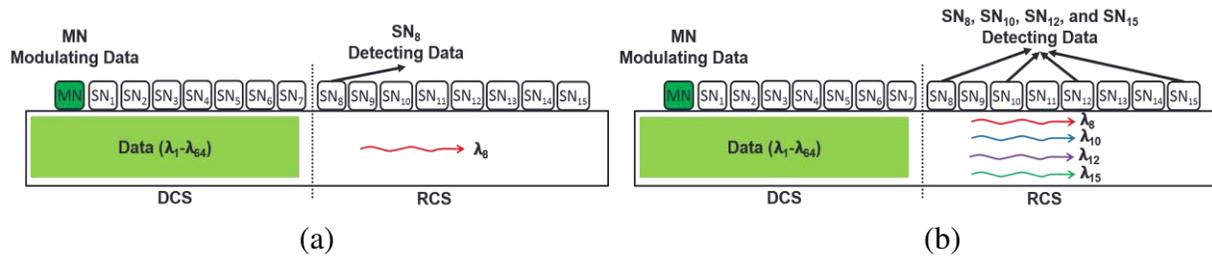


Figure 22 (a) Transmission of unicast data from an MN to SN₈ in *MSNoC*, which shows receiver selection wavelength λ_8 in RCS of the SWMR waveguide; (b) Multicast of data from an MN to multiple SNs SN₈, SN₁₀, SN₁₂, and SN₁₅ in *MSNoC*, which shows respective receiver selection wavelengths λ_8 , λ_{10} , λ_{12} , and λ_{15} in RCS of the SWMR waveguide.

Interestingly, the laser power injected in the SWMR waveguide group for multicasting in an *MSNoC* does not change with the number of nodes that need to receive the multicast message. We designed the laser source for the worst-case power loss, which occurs when all of the SNs receive a multicast message (i.e., broadcast message) from an MN. We have considered this extra laser power overhead when presenting energy-delay-product and energy-per-bit results for the *MSNoC* cluster in our experimental results section. In this chapter, we do not consider optimizing laser power through a laser power management scheme. However, it is possible to integrate previously proposed laser power management schemes [14], [39], as these works are orthogonal to our work.

Figure 22(a) and (b) illustrate the difference between transmission of unicast and multicast messages in our *MSNOC* cluster. Suppose an MN needs to multicast data to SN₈, SN₁₀, SN₁₂, and SN₁₅ whose corresponding receiver selection wavelengths are λ_8 , λ_{10} , λ_{12} , and λ_{15} , respectively. The MN modulates in the next RCS, such that only λ_8 , λ_{10} , λ_{12} , and λ_{15} are made available by removing all the wavelengths except λ_8 , λ_{10} , λ_{12} , and λ_{15} (using the MN's modulators; Figure 22(b)) from the first SWMR waveguide of SWMR waveguide group. At the receiver end at SN₈, SN₁₀, SN₁₂, and SN₁₅, the detectors for wavelengths λ_8 , λ_{10} , λ_{12} , and λ_{15} respectively on the first SWMR waveguide of the SWMR waveguide group are switched on when these SNs are in the RCS. At SN₈, once λ_8 is detected in the receiver selection slot, the node prepares to receive data from all of the four SWMR waveguides within the SWMR waveguide group in the next DCS by partially detuning the ring detectors (partial detuning of ring resonators is employed to receive both unicast and multicast data in SN₈) from their corresponding resonating wavelengths in that node. The partial de-tuning of ring detectors of SN₈ will remove a portion of light available in the SWMR waveguide, leaving the remaining portion of light for the other detectors to absorb. Similarly, on detection of λ_{10} , λ_{12} , and λ_{15} , nodes SN₁₀, SN₁₂, and SN₁₅ respectively prepare to receive data in the next DCS. Note that our architecture does not differentiate between unicast and multicast transmissions, as it always employs partial detuning to receive both unicast and multicast messages.

3.3.2. SN-TO-MN COMMUNICATION IN MSNOC CLUSTER

All the SNs send data back to an MN in the aggregation phase, for which our *MSNOC* uses a Multiple-Write-Single-Read (MWSR) waveguide group for SN-to-MN communication, with each waveguide group having four MWSR waveguides. As shown in Figure 20(a), this MWSR

waveguide group starts from the last SN (i.e., SN₁₅) and traverses all of the remaining SNs (i.e., SN₁-SN₁₄) and finally terminates at the MN. In contrast to the SWMR waveguide group, all SNs have the ability to write on the MWSR waveguide group using their ring modulators (see Figure 20(c) which shows modulators of an SN on an MWSR waveguide) and the MN has the ability to read from the MWSR waveguide group using its ring detectors (see Figure 20(b) which shows detectors of an MN on an MWSR waveguide).

As all SNs are capable of modulating (writing) in an MWSR waveguide group, there is a need for arbitration between SNs to ensure that the data from different SNs does not destructively overlap on the shared MWSR waveguide group. We use a centralized electrical arbiter to avoid contention between SNs when writing to an MWSR waveguide group. This arbiter uses a round-robin arbitration scheme. However, by virtue of being a centralized arbiter, it lacks scalability beyond a certain cluster size. We address this drawback of the centralized arbiter in Section 5. Furthermore, MSNoC exploits the centralized arbiter to enable flow control in the SN-to-MN communication. We employ an Xon/Xoff flow control mechanism to control packet flow from an SN to MN. Whenever, the receiving buffer in the MN is full then a signal is sent to the centralized arbiter, such that this arbiter stops assigning MWSR waveguide groups to the SNs. Otherwise, if the buffer is not full then the centralized arbiter allocates MWSR waveguide groups to SNs to transmit packets to MNs. As per the explanation provide in Section 3.3, a power waveguide (see Figure 20(a)) that runs in parallel with the MWSR waveguide group uses a series of splitters to supply photonic signals to the ring modulators to write data on to the MWSR waveguide group. As each of four MWSR waveguides within this MWSR waveguide group carries 64 wavelengths, therefore, each MWSR waveguide group requires 256 modulators and 256 detectors in the SN and

MN to write and read data, respectively. The total amount of photonic hardware required for the MSNoC architecture is quantified in Section 3.6.

3.3.3. SN-TO-SN COMMUNICATION IN MSNOC CLUSTER

SN-to-SN communication occurs in the *MSNoC* when the execution of high-performance data analytics applications is in the ‘shuffle’ phase. Our *MSNoC* enables SN-to-SN communication via the MN. We illustrate this SN-to-SN communication with a simple example. When SN₁₅ wants to send data to SN₅, first SN₁₅ sends data to the MN using an MWSR waveguide group, and then the MN sends the received data to SN₅ using an SWMR waveguide group. We show the SN₁₅-to-SN₅ communication path in Figure 20(a) as a dotted line. This process thus involves two O/E (optical to electrical) and two E/O (electrical to optical) conversions for each SN-to-SN transfer. The next section presents a performance analysis for an *MSNoC* cluster with different SN counts. In Section 3.5, we describe how multiple *MSNoC* clusters are combined to form the *BiGNoC* architecture.

3.4. MSNOC: SENSITIVITY ANALYSIS

In an *MSNoC* cluster, with the increase in number of SNs, contention between SNs to access an MWSR waveguide group increases. One possible solution to reduce this contention is to increase the number of MWSR waveguide groups in the *MSNoC* cluster. To understand the impact of this change, we performed a sensitivity analysis by varying the number of MWSR waveguide groups within an *MSNoC*, for different cluster sizes (8, 16, 32 nodes; each cluster has 1 MN and the remainder of the nodes are SNs). We modeled and simulated these variants of *MSNoC* at a cycle-accurate granularity with a SystemC-based NoC simulator. We considered three

applications: Text Mining [83], Financial Time Series [84], and Airline Query Processing [85]. The goal with these workloads was to emulate an environment with different intensities of MN-to-SN, SN-to-MN, and SN-to-SN traffic with diverse bandwidth needs.

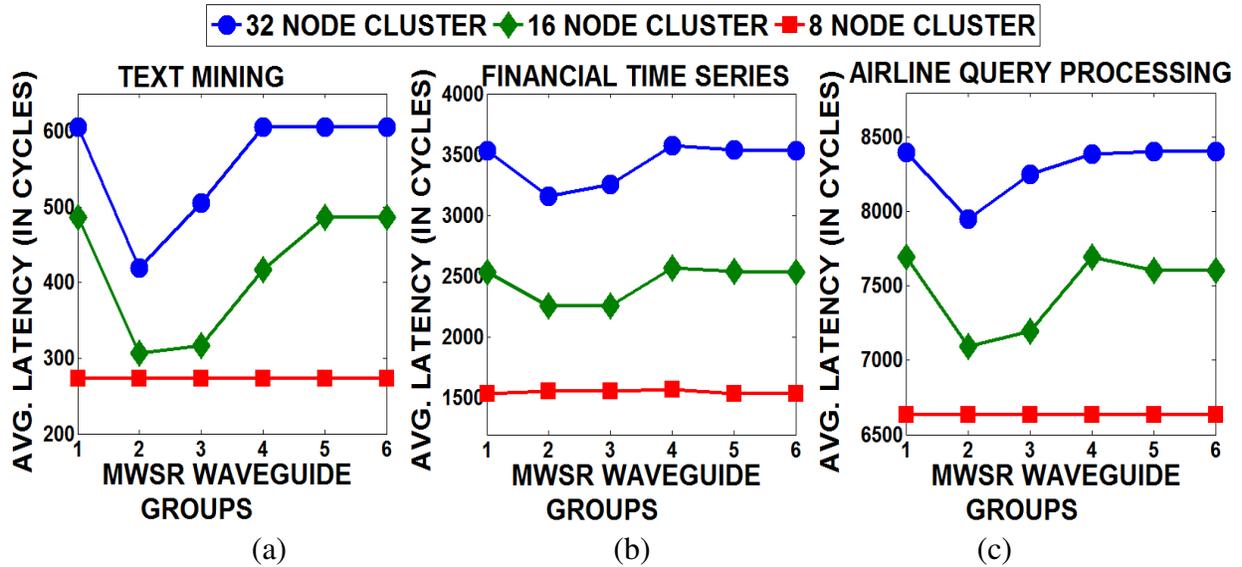


Figure 23 Variation of average packet latency in *MSNoC* cluster with (a) 32 nodes (b) 16 nodes, and (c) 8 nodes having different MWSR waveguide groups (each group has 4 waveguides) across three big data applications.

Figure 23 (a)-(c) show the variation of average packet latency with increase in number of MWSR waveguide groups (x-axes) for the three sizes of the *MSNoC* cluster, across the three big data applications. It can be observed that for a specific MWSR waveguide group count within an *MSNoC*, increase in cluster size (i.e., increase in node count) increases the average packet latency for all big data applications. Increase in number of nodes within a cluster increases contention between SNs to access the MWSR waveguide groups while sending data to an MN, which increases packet wait time in the buffers of SNs and ultimately increases overall packet latency. From Figure 23 (a)-(c), it can also be seen that with the increase in MWSR waveguide groups, the average packet latency first decreases until the waveguide group count reaches two. When MWSR

waveguide group count is increased beyond two, the latency starts increasing. Intuitively, increase in number of MWSR waveguide groups from one to two increases the SN-to-MN data rate (as two MWSR waveguide groups enable two packets to be sent simultaneously from two SNs to an MN), which decreases packet waiting time in the buffers of SNs and reduces the average packet latency. Despite the increase in data rate from SN-to-MN, with the increase in number of MWSR waveguide groups beyond two, there is saturation in the data channel to the MN (as this data channel is capable of sending only one packet per cycle from the concentrator to a master core). This increases the waiting time of packets at the receiving buffers of MGIs and increases average packet latency across all the big data applications.

Based on the analysis presented above, we optimally select two MWSR waveguide groups for *MSNoCs* with cluster sizes of 32 and 16 nodes. Additionally, from the Figure 23(a)-(c) it can also be seen that average latency for an *MSNoC* with 8 nodes remains constant for all MWSR waveguide group counts across all the benchmark applications. From this result, it can be concluded that in an *MSNoC* with 8 nodes, a single MWSR waveguide group is sufficient and optimal for SN-to-MN communication. We use these optimally determined MWSR waveguide group counts for different cluster sizes in our homogeneous and heterogeneous master-servant multi-cluster architecture (*BiGNoC*) which we describe in detail in the next section.

3.5. BIGNOC ARCHITECTURE

3.5.1. HOMOGENEOUS BIGNOC ARCHITECTURE

In Section 3.3, we presented an *MSNoC* architecture that aims to effectively connect an MN and many SNs within a master-servant cluster using MWSR and SWMR waveguide groups. Typically, large-scale data analytics applications require a greater number of servant cores than

can be accommodated in a single *MSNoC* cluster. There are two ways to address the requirement for additional servant cores: increase the cluster size or use multiple inter-connected clusters. We prefer the latter solution as increase in cluster size leads to: (i) increase in power dissipation of the SWMR and MWSR waveguide groups (see Table 8 later in the chapter), (ii) increase in average packet latency (see Figure 23), and (iii) increase in MWSR waveguide group arbiter complexity. These drawbacks suppress the power and performance benefits of photonic interconnects. Moreover, increase in cluster size limits the number of available masters within a cluster as the *MSNoC* is designed to have only one master node. Therefore, we propose a homogeneous multi-cluster architecture (*BiGNoC-HOM*) with four uniform clusters represented as C_0 , C_1 , C_2 , and C_3 , as shown in Figure 24(a), where each cluster has 16 nodes (i.e., 64 cores).

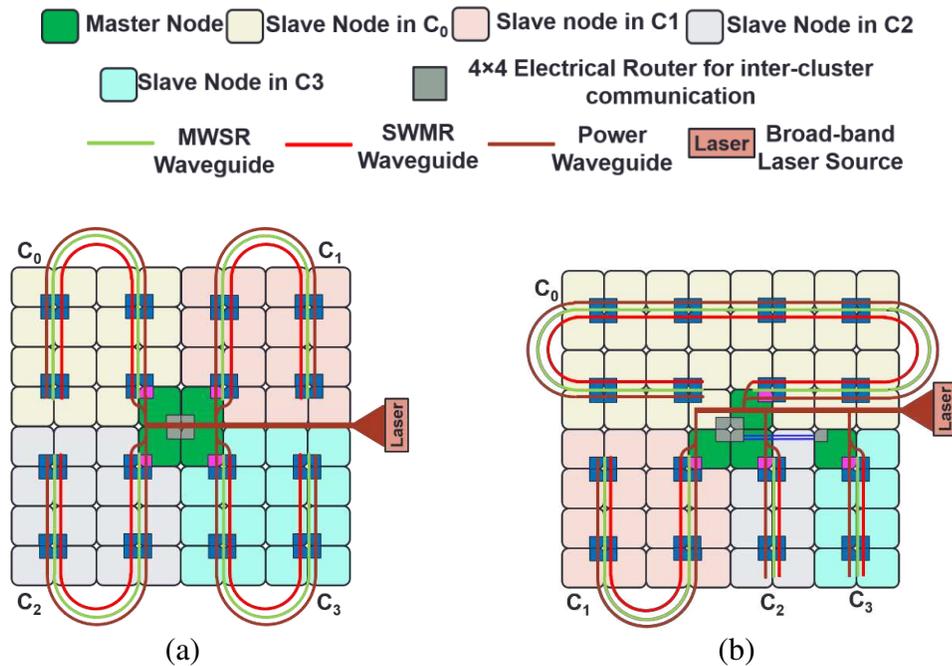


Figure 24 (a) Homogeneous *BiGNoC* with four uniform clusters C_0 , C_1 , C_2 , C_3 , with each cluster having 16 nodes, (b) Heterogeneous *BiGNoC* with four clusters C_0 , C_1 , C_2 , and C_3 having 32, 16, 8, and 8 nodes, respectively.

Each 16-node cluster in the *BiGNoC-HOM* architecture uses one SWMR waveguide group for MN-to-SN communication. As explained in Section 3.3.1, each SWMR waveguide group is divided into two time slots to enable receiver selection. Furthermore, based on the sensitivity analysis presented in the previous section, we optimally select two MWSR waveguide groups in each cluster for SN-to-MN communication. This architecture considers a single broadband laser source to power all of its SWMR and MWSR waveguides and uses 64 wavelengths in each waveguide for data communication. We add three more splitters to the power waveguide, to distribute laser power to the SWMR and MWSR waveguide groups of the four clusters in *BiGNoC-HOM*.

Each MN has a memory controller to send and receive data from off-chip main memory with dedicated channels for communication. Therefore, *BiGNoC-HOM* uses four memory controllers, where each is associated with an MN within a cluster. In addition, as shown in Figure 24, all the four MNs within the four clusters of *BiGNoC-HOM* are connected to a single 4×4 electrical router using their external electrical I/O ports (shown at the top left of Figure 20(a)). This electrical router is used for inter-cluster communication. We have considered a four-stage pipelined electrical router with 4 I/O ports that are connected to four MNs with the following pipeline stages: buffer write/route computation, region validation/switch allocation, switch traversal, and link traversal. This router has an input and output queued crossbar and uses double buffering with an 8-flit buffer size to more effectively cope with the higher photonic path throughput. Each master node is provisioned with an additional buffer which receives and stores packets from other clusters.

Intuitively, inter-cluster MN-to-MN communication occurs in one hop through the electrical router. Inter-cluster MN-to-SN and SN-to-MN communication require two hops: inter-cluster MN-to-SN communication requires MN-to-MN (inter-cluster) and MN-to-SN (intra-cluster) hops,

whereas inter-cluster SN-to-MN communication requires SN-to-MN (intra-cluster) and MN-to-MN (inter-cluster) hops. Further, inter-cluster SN-to-SN communication requires three hops: SN-to-MN (intra-cluster), MN-to-MN (inter-cluster), and MN-to-SN (intra-cluster). We illustrate the SN-to-SN communication across different clusters with a simple example. If node N_2 (i.e., SN) of C_0 needs to send a packet to node N_{10} (i.e., SN) of cluster C_1 , then N_2 of C_0 first sends data to N_0 (i.e., MN) of C_0 using an MWSR waveguide group. Then from this node the packet is sent to N_0 (i.e., MN) of C_1 through the electrical router that enables inter-cluster communication. Lastly, the packet is sent to N_{10} of C_1 using the SWMR waveguide group in that cluster. Thus, inter-cluster SN-to-SN communication incurs minimal overhead with only two O/E and two E/O conversions, which is similar to intra-cluster SN-to-SN communication.

3.5.2. HETEROGENEOUS BIGNOC ARCHITECTURE

As explained in the previous subsection, *BiGNoC-HOM* with four uniform clusters can enable inter-cluster communication between MNs and SNs. While executing applications with larger servant core count requirements, *BiGNoC-HOM* incurs higher inter-cluster traffic. This increase in inter-cluster traffic via slower electrical links may reduce the performance of the proposed *BiGNoC-HOM* architecture. This motivates us to design a heterogeneous version of *BiGNoC* (*BiGNoC-HET*) with four clusters, but with different cluster sizes.

In *BiGNoC-HET*, we use clusters C_0 , C_1 , C_2 , and C_3 with 32, 16, 8, and 8 nodes, respectively, as shown Figure 24(b). To enable receiver selection in SWMR waveguide groups of these clusters, we divided the waveguides in clusters C_0 , C_1 , C_2 , and C_3 into 4, 2, 1, and 1 time slots respectively, based on the time taken by light to traverse these waveguides on a die. Based on the sensitivity analysis presented in Section 3.4, we use 2, 2, 1, and 1 MWSR waveguide groups for clusters C_0 ,

C_1 , C_2 , and C_3 respectively. Similar to *BiGNoC-HOM*, we use four memory controllers to control off-chip memory and an electrical router to connect all four clusters of *BiGNoC-HET*.

In *BiGNoC* (especially *BiGNoC-HET*), scheduling of applications plays a crucial role in enhancing overall performance. For example, *BiGNoC-HET* can achieve better performance when an application with a greater servant core requirement is scheduled to a cluster with more servant cores. In contrast, scheduling a larger application on multiple smaller clusters will increase inter-cluster communication, which in turn may degrade performance. This motivates us to design an application scheduling algorithm for *BiGNoC* which is presented in the next subsection. We perform a detailed comparative study between *BiGNoC-HOM* and *BiGNoC-HET* in Section 3.6.3.

Algorithm 1 Application scheduling in *BiGNoC*

Inputs: Applications (AP_i) with master cores (MA_i) and servant cores (SA_i) requirements, and *BiGNoC* with clusters (C_j), master cores (MC_j), and servant cores (SC_j)

```

1:  Sort  $AP_i$  (highest SA to lowest SA)
2:  Sort BiGNoC clusters (highest SC to lowest SC)
3:  for all  $i$  do  $NSA_i = SA_i$ ;  $NMA_i = MA_i$ ;
4:  for all  $j$  do  $FSC_j = SC_j$ ;  $FMC_j = MC_j$ ;
5:  for each  $AP_i$  do
6:    for each  $C_j$  do
7:      if  $FSC_j > 0$  then // Checks for free cores in clusters
8:        if  $FSC_j - NSA_i \geq 0$  then
9:          Do_ Scheduling ( $AP_i \rightarrow NSA_i$  servant cores of  $C_j$ ) //Map servants
10:          $FSC_j = FSC_j - NSA_i$ ;  $NSA_i = 0$ ;
11:         if  $FMC_j > 0$  and  $FMC_j - NMA_i \geq 0$  then
12:           Do_ Scheduling ( $AP_i \rightarrow NMA_i$  master cores of  $C_j$ )//Map masters
13:            $FMC_j = FMC_j - NMA_i$ ;  $NMA_i = 0$ ;
14:         else if  $FMC_j > 0$  and  $FMC_j - NMA_i < 0$  then
15:           Do_ Scheduling ( $AP_i \rightarrow (NMA_i - FMC_j)$  master cores of  $C_j$ )
16:            $NMA_i = NMA_i - FMC_j$ ;  $FMC_j = 0$ ;
17:         else
18:           Do_ Scheduling ( $AP_i \rightarrow (NSA_i - FSC_j)$  servant cores of  $C_j$ )
19:            $NSA_i = NSA_i - FSC_j$ ;  $FSC_j = 0$ ;
20:         if  $FMC_j > 0$  and  $FMC_j - NMA_i \geq 0$  then
21:           Do_ Scheduling ( $AP_i \rightarrow NMA_i$  master cores of  $C_j$ )
22:            $FMC_j = FMC_j - NMA_i$ ;  $NMA_i = 0$ ;
23:         else if  $FMC_j > 0$  and  $FMC_j - NMA_i < 0$  then
24:           Do_ Scheduling ( $AP_i \rightarrow (NMA_i - FMC_j)$  master cores of  $C_j$ )
25:            $NMA_i = NMA_i - FMC_j$ ;  $FMC_j = 0$ ;

```

Output: Scheduled master-servant cores of app onto clusters of *BiGNoC*

3.5.3. APPLICATION SCHEDULING IN BIGNOC

Algorithm 1 shows the pseudo-code for the application scheduling procedure in *BiGNoC*. Applications (AP_i) are assumed to have master core (MA_i) and servant core (SA_i) requirements. The target *BiGNoC* platform is characterized by its clusters (C_j), master cores (MC_j), and servant cores (SC_j). First, the applications and *BiGNoC* platform clusters are sorted in the descending order of their SA_i and SC_j counts, respectively (steps **1-2**). In steps **3-4**, the algorithm initializes the required number of master cores (NMA_i) and servant cores (NSA_i) that are to be scheduled for each application, and also initializes the number of available free master cores (FMC_j) and free servant cores (FSC_j) in each cluster of *BiGNoC*, respectively. A nested loop iterates over all applications (AP_i) and clusters (C_j) in steps **5-6**. If FSC_j are available in cluster C_j at step **7**, then in steps **8-25**, we assign master and servant cores of *BiGNoC* to applications. We compare the number of available free servant cores within a cluster with the number of servant cores required by an application. If the number of free servant cores within a cluster are greater (steps **8-10**), then we assign the required free servant cores in the current cluster to the current application, else we assign all the free servant cores in the current cluster to the current application (steps **17-19**). For every free servant core assignment to an application in a cluster, we also compare the number of available free master cores within the cluster with the number of master cores required by an application. If the number of free master cores within a cluster are greater (steps **11-13** and **20-22**), then we assign the required free master cores in the current cluster to the current application, else we assign all the free master cores in current cluster to the current application (steps **17-19** and **23-25**). The proposed algorithm is used to schedule applications on both variants of *BiGNoC*.

3.6. EXPERIMENTS

3.6.1. EXPERIMENTAL SETUP

To evaluate the proposed *BiGNoC* architecture, we compared it with a traditional electrical mesh NoC (EMesh) and a broadcast optimized electrical mesh NoC (BO-EMesh) [98] as well as with two state-of-the-art photonic crossbar NoCs: Flexishare with token stream arbitration [13] and Firefly with a reservation assisted SWMR (R-SWMR) waveguide groups [12]. We modeled and simulated the NoC architectures at a cycle-accurate granularity with a SystemC-based NoC simulator for a 256-core CMP platform. We used this NoC simulator to emulate the execution of big data benchmarks across different architectures. In Flexishare, Firefly, BO-EMesh, and EMesh architectures with 256-cores, we have considered 16 master cores (similar to the number of master cores in *BiGNoC*; recall that *BiGNoC* has 4 MNs, which corresponds to 16 master cores) and the remaining cores are considered as servant cores for a fair comparison with the *BiGNoC* architecture. We used five big data benchmarks [82], [85]- [86] (Table 7) to create multi-application workloads. The goal with these workloads is to emulate an environment that executes future large-scale data analytics applications having different master and servant combinations with diverse bandwidth needs.

Table 7 shows the variants of big data benchmarks with different master-servant requirements considered for our analysis. We created 12 multi-application workloads from these benchmarks. Each workload combines 2 to 4 benchmarks, such that the summation of all the master cores and servant cores within the multi-application workload is lower than the number of available cores (i.e., 256) in the CMP. As an example, the T (1-40)-A (5-50)-F (2-100)-N (1-50) workload combines variants of Text Mining with 1-master and 40-servants (T (1-40)), Airline Query Processing with 5-masters and 50-servants (A (5-50)), Financial Time Series with 2-masters

and 100-servants (F (2-100)), and Netflix Movie Rating with 1-master and 50-servants (N (1-50)), and schedules them to clusters C_0 , C_1 , C_2 , and C_3 of *BiGNoC-HOM* and *BiGNoC-HET* using the application scheduling algorithm presented in Section 3.5.3. We analyzed the actual execution characteristics of the big data applications presented in Table 7 (such as the master processing time, servant processing time, etc.) that are measured using an Amazon’s Elastic Compute Cloud (EC2) instance [99], to generate traces that were fed into our network simulator. We set a “warm-up” period of 1M cycles and executed the applications for 100M cycles.

Table 7 Big Data application benchmarks, with three variants each, based on their master-servant requirements

Application	Representation	Application variants
Netflix Movie Rating	N (Masters-Servants)	N (1-50), N (1-70), N (1-100)
Text Mining	T (Masters-Servants)	T (1-40), T (1-60), T (1-80)
Gray Sort Contest	G (Masters-Servants)	G (5-200), G (7-200), G (10-200)
Financial Time Series	F(Masters-Servants)	F (2-100), F (3-110), F (4-120)
Airline Query Process	A (Masters-Servants)	A (5-50), A (5-60), A (5-70)

We targeted a 22nm process technology for the 256-core system. Based on geometric calculations of the waveguides for a 20mm× 20mm chip dimension, we estimated the time needed for light to travel in a photonic waveguide with a length of 12 cm from the first to the last node in a single pass of the MWMR waveguide group in Flexishare as 8 cycles (i.e., 1.6ns) at 5 GHz clock frequency. Throughout our analysis we use a flit size of 64 bits for BO-EMesh and EMesh and a total packet size of 512 bits for all PNoC architectures. We consider data modulation at both clock edges to enable simultaneous transfer of 512 bits in a single cycle, in the *BiGNoC-HOM*, *BiGNoC-HET*, Flexishare, and Firefly PNoCs. We considered an on-off switching time of 3.1 ps for a ring modulator and ring detector [13], which is less than one clock cycle (i.e., 200ps) at 5GHz frequency.

Table 8 Energy and Losses for Photonic Devices [73], [74], [100]

Cluster-wise static power per waveguide group of BiGNoC			
Waveguide Type	32-Node Power	16-Node Power	8-Node Power
P_{MWSR}	1.54W	0.62 W	0.21W
P_{SWMR}	5.72 W	2.69 W	1.26 W
Static power per waveguide group		Power	
$P_{SWMR-FY}$		1.15 W	
$P_{MWMR-FX}$		3.73 W	
Energy consumption type		Energy	
$E_{dynamic}$		0.42 pJ/bit	
$E_{logic-dyn}$		0.18 pJ/bit	
Photonic loss type		Loss (in dB)	
Microring through		-0.005	
Waveguide propagation per cm		-0.274	
Waveguide coupler/splitter		-0.2	
Waveguide bending loss		0.005 per 90 ⁰	

The static and dynamic energy consumption of the electrical routers is based on results obtained from the DSENT tool [75]. Energy consumption of various photonic components for all the photonic NoC architectures are adopted from photonic device characterizations in line with state-of-the-art proposals [73], [74], [100], and shown in Table 8. Here $E_{dynamic}$ is the energy per bit for modulators and photodetectors and $E_{logic-dyn}$ is the energy per bit for the driver circuits of modulators and photodetectors. $P_{SWMR-FY}$ and $P_{MWMR-FX}$ are the static power dissipation of SWMR and MWMR waveguide groups in Firefly and Flexishare architectures, respectively. Further, the P_{MWSR} and P_{SWMR} rows in Table 8 show static power dissipation of MWSR and SWMR waveguide groups of clusters in *BiGNoC* with sizes 32, 16, and 8 nodes, respectively. Also, we calculate power dissipation overheads of 75mW, 35mW, and 15mW in the electrical circuits of the SWMR waveguide groups in clusters of *BiGNoC* with sizes 32, 16, and 8 nodes, respectively, to realize partial detuning based on estimates from prior work [71]. All the static power dissipation values for waveguides presented in Table 8 include the power overhead of MR thermal tuning. We consider an MR heating power of 15 μ W per MR and detector responsivity of 0.8 A/W [74]. To

compute laser power dissipation, we calculated photonic loss in components, which sets the photonic laser power budget and correspondingly the electrical laser power. Lastly, based on our gate-level analysis, we estimate area overheads of 0.0065mm^2 and 0.008mm^2 , and power overheads of 0.12W and 0.16W in the electrical arbiters for the MWSR waveguide groups of *BiGNoC-HOM* and *BiGNoC-HET*, respectively.

3.6.2. BIGNOC: SENSITIVITY ANALYSIS

Our first set of experiments presents a sensitivity analysis to explore the optimal buffer size of the electrical router that is used for inter-cluster communication in two variants of our *BiGNoC* architecture with 256 cores: *BiGNoC-HOM* and *BiGNoC-HET*. *BiGNoC-HOM* has four homogeneous clusters with each cluster having 16 nodes; and *BiGNoC-HET* has four clusters with 32, 16, 8, and 8 nodes, respectively.

Figure 25(a) and (b) show the average packet latency for three multi-application big data workloads on *BiGNoC-HOM* and *BiGNoC-HET*, with buffer depth of the electrical router varying from 8 to 40. In this analysis, to compute average packet latency we have considered the delay incurred by the packet to move from the source node to the destination node along with the queuing delays in routers and interfaces. The three workloads were chosen to possess high, medium, and low aggregate inter-cluster traffic, to explore the impact of application traffic on buffer depth. We characterized inter-cluster traffic of an application by counting the number of transfers through the electrical router, which is used for inter-cluster communication.

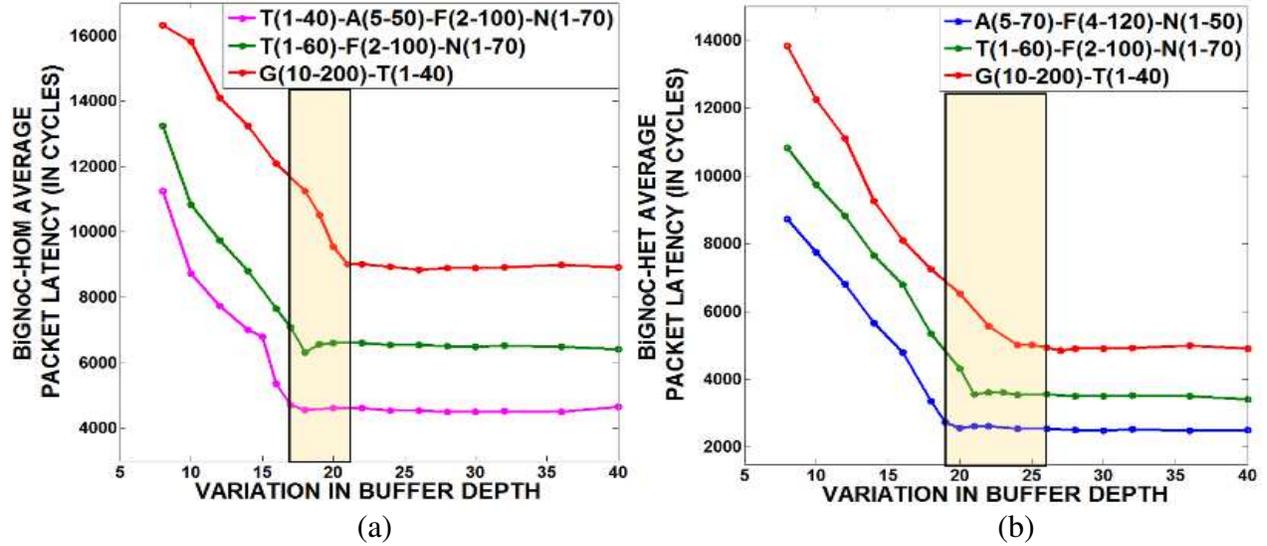


Figure 25 Average packet latency comparison for (a) *BiGNoC-HOM* and (b) *BiGNoC-HET* in a 256-core CMP with different buffer depths (8-40).

At a particular buffer depth for both *BiGNoC-HOM* and *BiGNoC-HET*, Figure 25 shows higher average packet latency for workloads with higher inter-cluster traffic (i.e., G(10-200)-T(1-40)) compared to workloads with lower inter-cluster traffic (i.e., T(1-40)-A(5-50)-F(2-100)-N(1-50) for *BiGNoC-HOM* and A(5-70)-F(4-120)-N(1-50) for *BiGNoC-HET*) as queuing of packets occurs at the master nodes for workloads with higher inter-cluster traffic, which increases their queuing delay and average packet latency. Also, for all workloads executing on both *BiGNoC-HOM* and *BiGNoC-HET*, a smaller buffer size should intuitively result in higher average packet latency, as the buffer in the electrical router becomes more frequently full and creates back pressure on the buffers in the MN of each cluster of *BiGNoC-HOM* and *BiGNoC-HET*. As a result, the centralized arbiter within each cluster stops assigning MWSR waveguide groups to SNs (due to Xon/Xoff flow control mechanism used within each cluster; for explanation see Section 3.3.2) in that cluster, which are used to transfer packets to MN, which in turn increases packet queuing delay within each SN and incurs higher average packet latency.

On the other hand, beyond a particular buffer depth in both *BiGNoC-HOM* and *BiGNoC-HET* the average packet latency of all the applications saturate. After a particular buffer depth, the buffer in the electrical router of both variants of BiGNoC seldom gets full, which is the main reason for this saturation. A careful observation of the plots in Figure 25 shows that for workloads with lower inter-cluster traffic (i.e., T(1-40)-A(5-50)-F(2-100)-N(1-50) for *BiGNoC-HOM* and A(5-70)-F(4-120)-N(1-50) for *BiGNoC-HET*) latency saturation occurs at a small buffer depth, whereas for workloads with higher inter-cluster traffic (i.e. G(10-200)-T(1-40) for both *BiGNoC-HOM* and *BiGNoC-HET*) latency saturation occurs at a large buffer depth. However, as shown in Figure 25(a) and (b), there is a region (light yellow shaded region) between saturation points of low inter-cluster traffic application and high inter-cluster traffic application, where both *BiGNoC-HOM* and *BiGNoC-HET* archive optimal performance. Therefore, we chose to use 21 and 26 as the optimal buffer depth for *BiGNoC-HOM* and *BiGNoC-HET*, respectively, which are the highest buffer depths of the optimal performance regions shown in Figure 25(a) and (b). We use these optimal buffer depths for *BiGNoC-HOM* and *BiGNoC-HET* in the rest of our analysis.

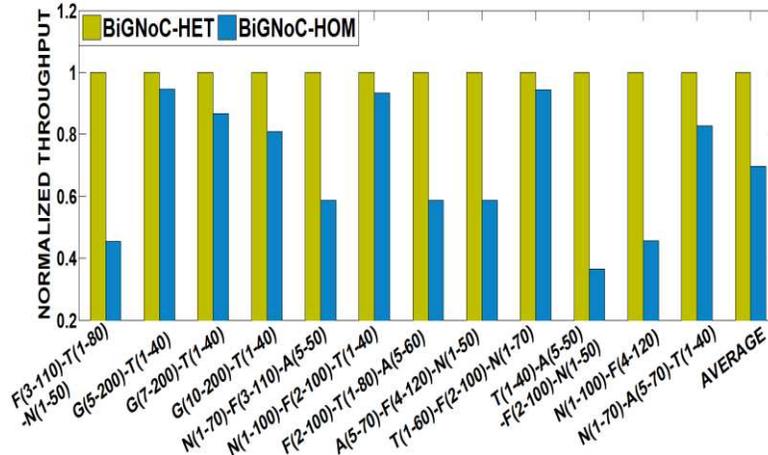
3.6.3. EXPERIMENTAL RESULTS

Our next set of experiments presents a comparative study between *BiGNoC-HOM* and *BiGNoC-HET*. We used the optimal buffer depth of 21 and 26 for *BiGNoC-HOM* and *BiGNoC-HET*, respectively (determined as per the previous subsection) in this comparative study. Figure 26(a) and (b) present detailed simulation results that quantify the average throughput and energy-delay product (EDP) for *BiGNoC-HOM* and *BiGNoC-HET*, for twelve multi-application workloads. Results are normalized with respect to the *BiGNoC-HET* results.

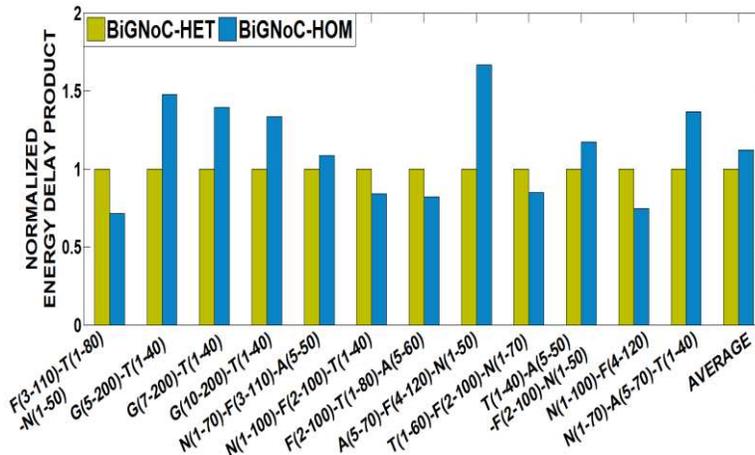
From Figure 26(a) it can be seen that on an average *BiGNoC-HET* has 30.4% higher average throughput compared to *BiGNoC-HOM*. Variable cluster sizes in *BiGNoC-HET* help reduce the inter-cluster traffic while executing big data workloads involving different master-servant combinations. This decrease in inter-cluster traffic improves utilization of MWSR and SWMR waveguides within a cluster and increases the throughput of *BiGNoC-HET* compared to *BiGNoC-HOM*. Also, from Figure 26(b) it can be observed that on an average *BiGNoC-HET* has 12.5% lower EDP compared to *BiGNoC-HOM*. Decrease in average latency and decrease in trimming energy (due to decrease in number of detectors) decreases EDP of *BiGNoC-HET* compared to *BiGNoC-HOM* even though there is increase in laser energy for *BiGNoC-HET*. However, from Figure 26(b) it can also be seen that for a few application combinations, EDP of *BiGNoC-HET* is higher compared to *BiGNoC-HOM*. For these application combinations, *BiGNoC-HET* achieves lower average latency benefits compared to *BiGNoC-HOM*, which increases *BiGNoC-HET*'s EDP (as *BiGNoC-HET* always consumes more laser energy than *BiGNoC-HOM*). From the average throughput and EDP results presented in Figure 26, we can summarize that *BiGNoC-HET* achieves better performance with lower EDP compared to *BiGNoC-HOM*, which motivates its usage towards executing future large-scale data analytics applications. Therefore, for our next set of experiments we have used only *BiGNoC-HET* to estimate benefits over electrical and photonic NoC architectures from prior work.

In the next set of experiments, we compare network throughput, average packet latency, and energy-per-bit (EPB) of *BiGNoC-HET* with the EMesh, BO-EMesh, Flexishare with token stream arbitration [13], and Firefly with R-SWMR waveguide [12] architectures. Figure 27(a)-(c) show the results of this comparative analysis, where all the results are normalized with respect to the EMesh results. From the throughput comparison in Figure 27(a), it can be observed that, not

surprisingly, *BiGNoC-HET* provides 8.7 \times and 7.2 \times higher throughput than EMesh and BO-EMesh, respectively, due to the presence of higher bandwidth photonic links for data communication.



(a)



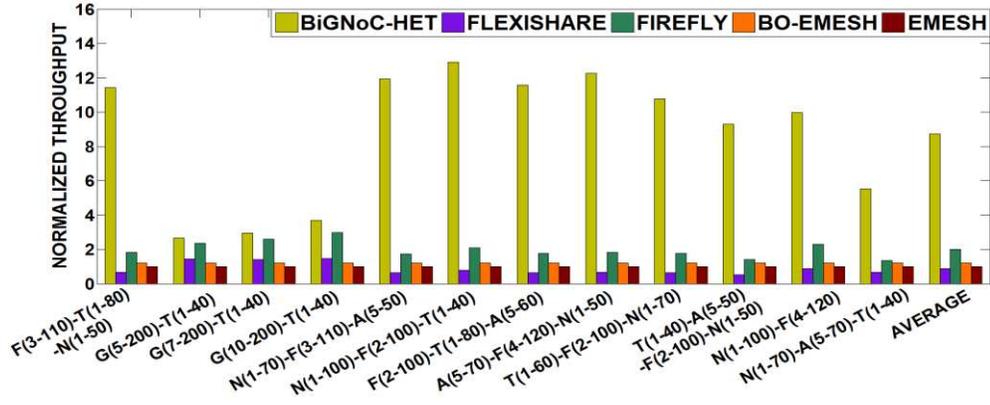
(b)

Figure 26 (a) Normalized throughput, (b) normalized EDP comparison of *BiGNoC-HOM* with *BiGNoC-HET* for 256-core CMP. Results are shown for multi-application workloads and normalized w.r.t. *BiGNoC-HET*.

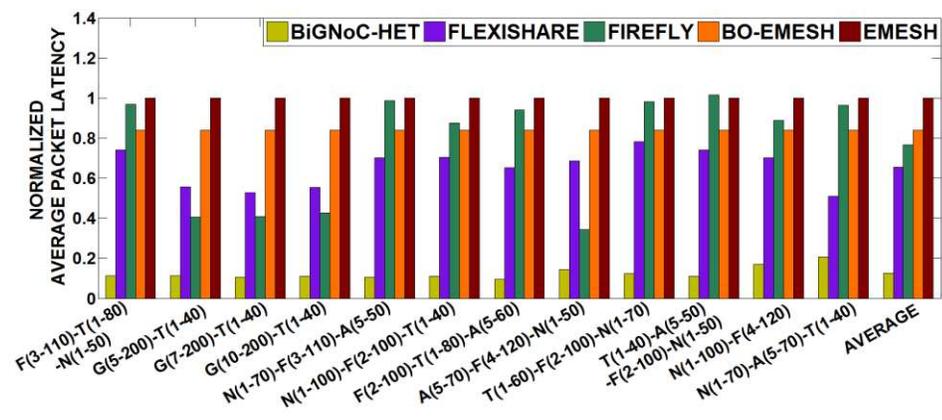
BiGNoC-HET has nearly 9.9 \times greater throughput compared to Flexishare. Even though Flexishare uses MWMM waveguides and time division multiplexing (TDM), its token stream

arbitration reduces its waveguide utilization and overall throughput compared to *BiGNoC-HET*. In Flexishare, arbitration wavelengths corresponding to MWMR data waveguides are injected serially into the arbitration waveguide and a node that grabs a token in the arbitration waveguide gets exclusive access to the corresponding MWMR data waveguide, which limits Flexishare's ability to perform simultaneous data transfers. In contrast, *BiGNoC-HET* has dedicated photonic paths (MWSR waveguide group for SN-to-MN communication and SWMR waveguide group for MN-to-SN communication) between the master node and servant nodes within each cluster. This helps in increasing simultaneous data transfers in *BiGNoC-HET* with increase in number of clusters. *BiGNoC-HET* also facilitates efficient multicasting to improve throughput over Flexishare by using its SWMR waveguide groups from MN to SNs, whereas in Flexishare, multiple unicast packets are sent from the master core to servant cores instead of a single multicast packet.

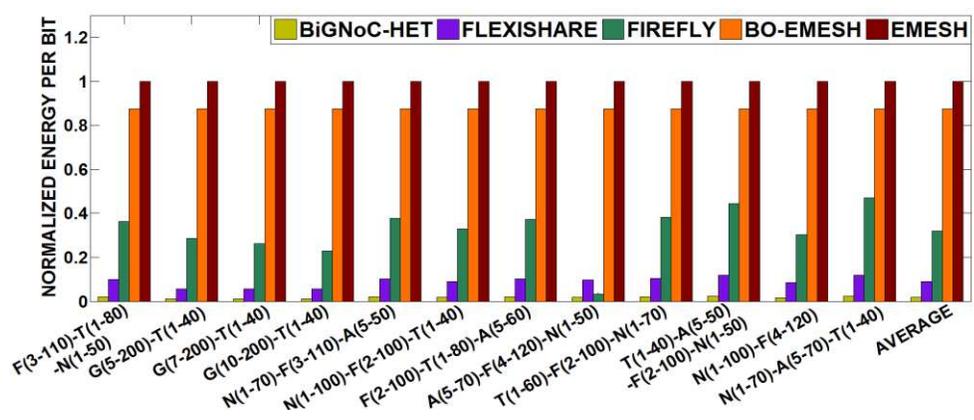
BiGNoC-HET has 4.4× higher throughput compared to Firefly. This is due to the near light speed communications for a majority of the path traversed by the data in *BiGNoC-HET* using photonic links, whereas Firefly being a hybrid network, utilizes slower electrical links for a significant portion of the path traversed by the data. These mechanisms also improve the average packet latency in *BiGNoC-HET*, as shown in Figure 27(b), by reducing the time spent waiting for access to the photonic waveguides. On average *BiGNoC-HET* has 81%, 84%, 85%, and 88% lower average packet delay over Flexishare, Firefly, BO-EMesh, and EMesh, respectively for the different multi-application workloads.



(a)



(b)



(c)

Figure 27 Normalized (a) throughput (b) latency (c) EPB comparison of *BiGNoC-HET* with other architectures for a 256-core CMP. Results are for multi-application workloads and normalized w.r.t. EMesh.

Figure 27(c) shows the EPB comparison between the architectures. It can be observed that on average *BiGNoC-HET* has 88%, 90%, 96%, and 98% lower EPB compared to Flexishare, Firefly, BO-EMesh, and EMesh, respectively. *BiGNoC-HET* has lower EPB compared to BO-EMesh and EMesh, as it uses energy efficient photonic links for data transfer instead of power hungry electrical links. Most of the energy in the photonic architectures was consumed in the form of static energy.

Table 9 Photonic Hardware Comparison

Architecture	Waveguides	Modulators	Detectors
BiGNoC-HOM	12	31,744	17,408
BiGNoC-HET	10	33,280	11,776
Flexishare	33	131,080	131,648
Firefly	64	4,096	28,672

Table 9 shows the photonic hardware comparison between the PNoC architectures. It can be seen that *BiGNoC-HET* has 82% less photonic hardware compared to Flexishare. This reduction in photonic hardware reduces its overall static energy consumption and its EPB. Although both *BiGNoC-HET* and Firefly use multicasting in their SWMR waveguides, the lower EPB of *BiGNoC-HET* compared to Firefly is due to the higher energy consumption in the electrical network of the Firefly architecture.

3.7. CONCLUSIONS

We presented a new application-specific *BiGNoC* architecture that features master-servant clusters with efficient utilization of SWMR and MWSR waveguides to improve performance while executing large-scale data analytics applications. *BiGNoC* exploits efficient multicasting in photonic waveguides to achieve high data rates. In particular, we showed how *BiGNoC-HET*, a

variant of *BiGNoC*, improves performance due to improved photonic channel utilization and its ability to adapt to time-varying application performance goals while co-running multiple large-scale data analytics applications. *BiGNoC-HET* improves throughput by up to 9.9×, packet latency by up to 88%, and energy-per-bit by up to 98% over traditional EMesh, broadcast optimized EMesh, and state-of-the-art photonic NoC architectures (Flexishare and Firefly). These results corroborate the excellent capabilities of our proposed *BiGNoC* architecture towards executing large-scale data analytics applications.

4. CROSSTALK MITIGATION FOR HIGH-RADIX AND LOW-DIAMETER PHOTONIC NOC ARCHITECTURES

PNoC architectures have shown the potential to replace electrical networks-on-chip as they can attain higher bandwidth with lower power-dissipation for on-chip communication. But microring-resonators, which are the basic building blocks of PNoCs, are highly susceptible to crosstalk that can notably degrade OSNR, reducing reliability in PNoCs. We propose two novel encoding mechanisms to improve worst-case-OSNR by reducing crosstalk noise in microring-resonators used within high-radix and low-diameter crossbar-based PNoCs. Our evaluation results indicate that the encoding schemes improve worst-case-OSNR in Corona and Firefly PNoCs by up to 18%.

4.1. MOTIVATION AND CONTRIBUTION

MRs suffers from intrinsic crosstalk-noise and power-loss due to their design imperfections. The crosstalk noise severely impacts PNoCs, especially crossbar architectures with high MR counts, where the generated crosstalk is intensified, leading to transmission errors. For example, the Corona [67] crossbar architecture has worst-case OSNR of 14dB [100] in its data channels, which is insufficient for reliable data communication, as its corresponding bit-error-rates (BER) are very high, in the order of 10^{-3} .

Crosstalk in DWDM-based PNoCs mainly occurs due to inefficient coupling in ring-detectors, with non-resonant-wavelengths closer to the detector resonance-wavelengths creating greater crosstalk-noise. In the electrical domain, crosstalk occurs when adjacent wires simultaneously transition in opposite directions. The code-words used in the electrical domain are

not directly applicable to the photonic domain. For example, forward-error-correcting (FEC) codes that are effective in correcting erroneous bit-flips in the electrical domain utilize code-words with adjacent 1's that cannot improve OSNR in the photonic domain. Thus different techniques to mitigate crosstalk noise and improve reliability are needed for PNoCs.

We observe that when transmitting data in PNoCs, crosstalk noise in MRs depends on the characteristics of data values propagating in the photonic waveguide. Therefore we propose two novel techniques to intelligently reduce undesirable data value occurrences in a photonic waveguide. These techniques are easily implementable in any existing DWDM-based photonic crossbar without requiring major modifications to the architectures, unlike previously proposed crosstalk mitigation techniques (e.g., [42]) that are targeted to reduce crosstalk in specific architectures by requiring modifications to their router designs. Our novel contributions in this chapter are:

- We design a crosstalk mitigation technique with 5-bit encoding (PCTM5B) to improve worst-case OSNR for DWDM-based photonic crossbar PNoCs;
- We introduce another crosstalk-mitigation scheme with 6-bit encoding (PCTM6B), that more aggressively improves OSNR but with relatively higher EDP overhead;
- We validate our schemes by implementing them on well-known crossbar PNoCs: Corona [67] and Firefly [12], for real-world multi-threaded PARSEC [43] benchmarks.

4.2. RELATED WORK

Several prior works have performed photonic crosstalk analysis at the device-level and architecture-level. The device-level efforts analyze crosstalk behavior for single waveguide crossings (e.g., [101]) and for one or few photonic switching elements with MRs [102]. Results

from these efforts show that crosstalk is very small at the device-level. However, at the architecture level, prior work (e.g., [42]) indicates that crosstalk has a significant impact on the OSNR of PNoCs because of the presence of several waveguide-crossings and switching-elements. Minimum OSNR was shown to be key limiting factor in the design of mesh-based PNoCs [42]. Further, [103] showed that in fat-tree-based PNoCs, crosstalk-noise power is higher than signal power when on-chip core counts exceed 128.

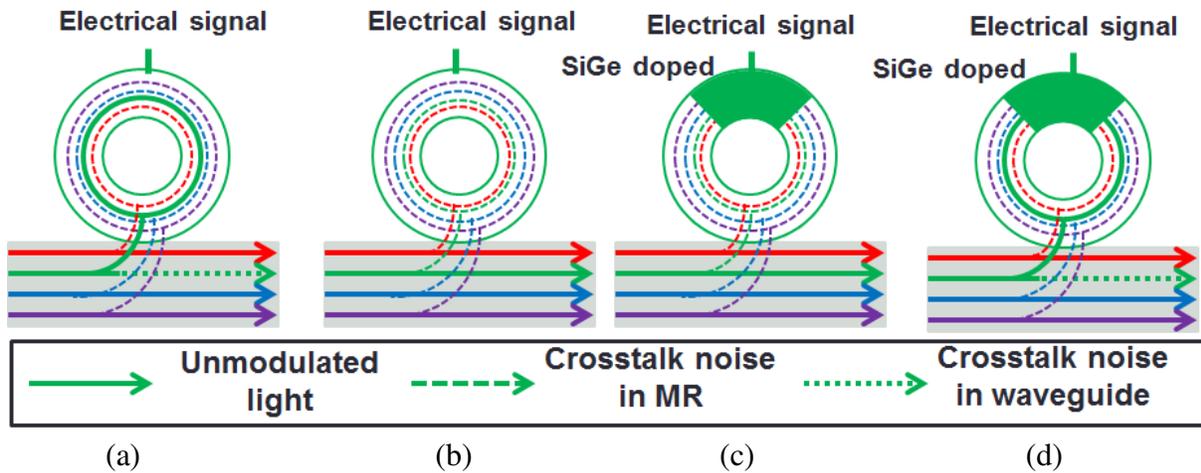


Figure 28 MR operation phases in DWDM-based waveguides (a) modulator modulating in resonance-wavelength (b) modulator in passing (through) mode (c) detector in passing-mode (d) detector in detecting-mode.

The above works focus on single-wavelength PNoCs, where crosstalk is generated from a single wavelength. A few prior works have also explored crosstalk in DWDM-based PNoCs where multiple wavelengths co-exist in a waveguide. A cascaded MR-based modulator structure is proposed in [102] for low-density DWDM waveguides, with an extinction ratio of 13dB and negligible crosstalk. In [104], losses in a similar multi-wavelength MR-based structure are measured. Though crosstalk appears negligible in these works where only four-wavelength DWDM waveguides are considered, in crossbar PNoC architectures such as Corona [67] that use

64-wavelength DWDM, there is significant crosstalk noise. The results in [100] demonstrate the damaging impact of crosstalk-noise in Corona, where the worst-case OSNR is estimated to be 14dB in data waveguides, which is insufficient for reliable data communication. A methodology to salvage network-bandwidth loss due to process-variation-drifts was proposed in [105], which reorders microrings and trims them to nearby wavelengths. In [106], [107], reliability aware multiple-segmented-bus (MSB) based PNoCs are proposed to enable data transfers with low BER. But this chapter does not address crosstalk reliability issues. Other efforts focus on architecture-specific crosstalk-mitigation [42], [103] by changing the physical design of PNoC routers. However, to date, no prior work has proposed generalized approaches to improve OSNR in an entire class of PNoC architectures, as we do in this chapter, for bit-parallel and packet-serial photonic data-transmission.

4.3. ANALYTICAL MODELS FOR CROSSTALK ANALYSIS IN DWDM-BASED PNO C ARCHITECTURES

4.3.1. OVERVIEW OF MR OPERATION IN DWDM-BASED PNOCS

DWDM-based PNoC architectures utilize photonic devices such as microring-resonators (MRs), photonic waveguides, splitters, and trans-impedance-amplifiers (TIAs). MRs in particular are essential to modulate light for transmission of data at a source-node (*data-modulation-phase*). MRs also detect light-modulated data from the waveguide at the destination-node (*data-detection-phase*) and subsequently generate proportional electrical signals that are amplified by TIAs. Each source node requires optical power/signals that are made available in the PNoC via power waveguides and splitters. An unfortunate property of silicon photonic waveguides is that signal propagation is lossy, i.e., the light signal is subject to losses such as through-loss, modulating-loss,

and detecting-loss in MRs, propagation-loss and bending-loss in waveguides, and splitting-loss in splitters. Such losses negatively impact OSNR in waveguides.

At any point in time in a photonic-waveguide, MRs are either in-resonance or out-of-resonance with respect to the incident wavelengths. In the resonance-mode, an MR couples light of a wavelength from the waveguide when its circumference is an integer multiple of that wavelength. Different-sized MRs in a DWDM-waveguide are thus required to simultaneously modulate data on different available wavelengths. These MRs in DWDM-based PNoCs suffer from intrinsic crosstalk-noise and power-loss.

Figure 28 (a)-(d) shows crosstalk noise (as dotted/dashed lines) in modulator and detector MRs during typical modulation/detection phases in the DWDM-waveguide. Whenever a modulator modulates a ‘0’ or a detector detects a ‘1’ from a particular wavelength (see Figure 28) by removing the light pulse, there is also crosstalk generated in the waveguide.

4.3.2. ANALYTICAL MODELS FOR CROSSTALK-NOISE AND SIGNAL-POWER

In this chapter, we consider crosstalk in DWDM-waveguides for the Corona PNoC architecture enhanced with token-slot arbitration [67] and the Firefly PNoC architecture [12]. In DWDM-based waveguides in both architectures, data-transmission requires modulating light using a series of MR-modulators equal to the number of wavelengths supported by DWDM. Similarly, data-detection at the receiver requires a group of MR-detectors equal to the number of DWDM wavelengths. We present analytical equations that model worst-case crosstalk-noise power, maximum power-loss, and OSNR in the MR-detector groups (similar equations are applicable to MR-modulator groups). We have validated these analytical models against device-level works [102]- [104]. In these analytical models we assume negligible inter-modulation

crosstalk. In the interest of brevity, we only present models and a description for the Corona PNoC, although our evaluation show results for both the Corona and Firefly PNoCs. We refer the reader to [12], [67] for details of the photonic crossbar topology and protocols employed in the Corona and Firefly PNoCs.

Table 10 Notations for photonic power-loss, crosstalk-coefficients and model-parameters [100]

Notation	Parameter type	Parameter value
L_P	Propagation loss	-0.274 dB per cm
L_B	Bending loss	-0.005 dB per 90°
L_{MI}	Inactive modulator through loss	-0.0005 dB
L_{MA}	Active modulator power loss	-0.6 dB
L_{DP}	Passing detector through loss	-0.0005 dB
L_{DD}	Detecting detector power loss	-1.6 dB
L_{S12}	1X2 splitter power loss	-0.2 dB
L_{S14}	1X4 splitter power loss	-0.2 dB
L_{S15}	1X5 splitter power loss	-0.2 dB
L_{S16}	1X6 splitter power loss	-0.2 dB
X_{MA}	Active modulator	-16 dB
X_{DD}	Detecting detector	-16 dB
Q	Q-factor of MR	9000
FSR	Free spectral range	62nm
Other model parameter notations		
$\Phi(i, j)$	Coupling factor between i^{th} microring resonators and j^{th} wavelengths in waveguide	
L	Photonic path length in cm	
B	Number of bends in photonic path	
λ_j	Resonance wavelength of MR	
R_{S12}	Splitting factor for 1X2 splitter	
R_{S14}	Splitting factor for 1X4 splitter	
R_{S15}	Splitting factor for 1X5 splitter	
R_{S16}	Splitting factor for 1X6 splitter	

The notations for parameters used in the analytical equations are shown in Table 10. Corona is designed for a 256-core single-chip platform, with cores grouped into 64 clusters, and 4 cores-per-cluster. For inter-cluster communication, Corona uses a photonic-crossbar topology with 64

data-channels. Each channel consists of 4 multiple-write-single-read (MWSR) waveguides with 64-DWDM in each waveguide. As modulation occurs on both positive and negative edges of the clock in Corona, 512-bits (cache-line size) can be modulated and inserted on 4-MWSR waveguides in a single cycle by a sender-node. A data-channel starts at a cluster called ‘home-cluster’, traverses other clusters (where modulators can modulate light in this channel and detectors can detect this light), and finally ends at the home-cluster again, at a set of detectors (optical termination).

A power-waveguide supplies optical power to each of the 64 data-channels at its home-cluster, through a series of 1×2 splitters starting from home-cluster 1 to 64. In each home-cluster, optical-power is distributed among 4-MWSR waveguides equally using a 1×4 splitter. As all 1×2 splitters are present before the last (64th) channel, this channel suffers the most signal-power-loss. Thus, the worst-case signal and crosstalk-noise power exists in the detector group of the 64th cluster node, and this node is defined as the worst-case power-loss-node (N_{WCPL}). For this node, signal-power ($P_{signal(j)}$) and crosstalk-noise-power ($P_{noise(j)}$) received at each detector j are expressed in Eq.(4) and (5) [100]:

$$P_{signal(j)} = L_{DD} \Phi(j, j) P_S(j, j) \quad (4)$$

$$P_{noise(j)} = L_{DD} P_N(j, j) + \sum_{i=1}^n \Phi(i, j) (P_S(i, j) + P_N(i, j)) \quad (i \neq j) \quad (5)$$

The parameters in the above equations are defined below:

$$P_S(i, j) = K_S \psi(i, j) P_{in}(i) \quad (6)$$

$$\psi(i, j) = \begin{cases} X_{DD} (L_{DP})^{j-1} & (j - 1 \geq i \text{ and } D_B = 1) \\ (L_{DP})^{j-1} & (j - 1 < i \text{ and } D_B = 1) \\ X_{MA} X_{DD} (L_{DP})^{j-1} & (j - 1 \geq i \text{ and } D_B = 0) \\ X_{MA} (L_{DP})^{j-1} & (j - 1 < i \text{ and } D_B = 0) \end{cases} \quad (7)$$

$$\Phi(i, j) = \frac{\delta^2}{((i-j)\frac{FSR}{n})^2 + \delta^2}, \text{ Here } \delta = \frac{\lambda_j}{2Q} \quad (8)$$

$$P_N(i, j) = \begin{cases} 0 & , \text{If } j > i \text{ and } D_B = 1 \\ K_N(L_{DP})^j P_{in}(i) & , \text{if } j \leq i \text{ and } D_B = 1 \\ 0 & , \text{If } j > i \text{ and } D_B = 0 \\ X_{MA} K_N(L_{DP})^j P_{in}(i) & , \text{if } j \leq i \text{ and } D_B = 0 \end{cases} \quad (9)$$

$$K_S = \begin{cases} (R_{S14})(L_{S14})(L_P)^L(L_B)^B(L_{MI})^{64 \times 63} (D_B = 1) \\ (R_{S14})(L_{S14})(L_P)^L(L_B)^B(L_{MI})^{64 \times 62 + 63} (D_B = 0) \end{cases} \quad (10)$$

$$K_N = \begin{cases} (R_{S14})(L_{S14})(L_P)^L(L_B)^B X_{MA}(L_{MI})^{64 \times 62 + 63} (D_B = 1) \\ (R_{S14})(L_{S14})(L_P)^L(L_B)^B X_{MA}(L_{MI})^{64 \times 62 + 62} (D_B = 0) \end{cases} \quad (11)$$

$P_S(i, j)$ in Eq.(6) is the signal-power of the i^{th} -wavelength received before the j^{th} -detector. Similarly in Eq.(9), $P_N(i, j)$ is the crosstalk-noise power of the i^{th} -wavelength before the j^{th} -detector. K_S and K_N in Eq.(10) and Eq.(11) represent signal and crosstalk-noise power-losses before the detector group of N_{WCPL} . $\psi(i, j)$ in Eq.(7) represents signal power-loss of the i^{th} -wavelength before the j^{th} -detector within the detector group of N_{WCPL} . $\Phi(i, j)$ in Eq.(8) is the crosstalk coupling-factor of the i^{th} -wavelength and the j^{th} -detector. Finally, we can define OSNR(j) of the j^{th} -detector of N_{WCPL} as the ratio of $P_{\text{signal}}(j)$ to $P_{\text{noise}}(j)$, as shown in Eq.(12):

$$OSNR(j) = \frac{P_{\text{signal}}(j)}{P_{\text{noise}}(j)} \quad (12)$$

These equations are sufficient to analyze signal and crosstalk-noise power during the detection of ones ($D_B='1'$) and zeros ($D_B='0'$) in a photonic waveguide. The next section uses these models to discuss how crosstalk mitigation techniques impact OSNR.

4.4. TECHNIQUES TO MITIGATE CROSSTALK NOISE

Crosstalk noise in the detectors of DWDM-based PNoCs is caused mainly due to inefficient coupling of MRs, as MR-detectors in their detecting mode not only couple photonic-power from their resonance-wavelengths but also couple some photonic-power from other wavelengths in the

waveguide. This coupling-factor (Φ) increases with a decrease in gap between resonant and non-resonant wavelengths of an MR-detector (Eq.(8)). Thus non-resonant wavelengths closer to the detector resonance wavelengths create greater crosstalk-noise. In addition, crosstalk-noise increases with increase in signal power of non-resonant wavelengths.

Based on the above observations, crosstalk noise can be mitigated by placing one or more ‘0’s adjacent to ‘1’s in the data in the waveguide, to reduce photonic signal-strength of immediate non-resonant wavelengths (adjacent wavelengths in DWDM). In this section, we present two techniques for mitigation of crosstalk noise in DWDM-based PNoCs that utilize this mechanism. The two techniques (PCTM5B, PCTM6B) employ 5-bit and 6-bit encoding for every 4-bit data block to reduce photonic-signal-strength of the immediate non-resonant wavelengths. The area/power/delay overheads of these techniques are discussed in Section 4.5.

4.4.1. PCTM5B ENCODING TECHNIQUE

Table 11 shows the 5-bit codes proposed in the *PCTM5B* scheme, to replace 4-bit data words. To implement this encoding technique on a 64-bit word, 16 additional bits are required, which increases the number of MRs by 25%. To facilitate simultaneous transfer of an entire packet in Corona, which requires 512-bits before encoding, we increase DWDM-degree in MWSR-waveguides from 64 to 65 and increase MWSR-waveguides in each channel from 4 to 5. To distribute optical-power between these waveguides, there is also a need to replace 1X4 splitters with 1X5 splitters. Therefore Eq.(10)-(11) for worst-case signal and crosstalk-noise power are changed to Eq.(13)-(14) below:

$$K_S = \begin{cases} (R_{S15})(L_{S15})(L_P)^L(L_B)^B(L_{MI})^{65 \times 63} (D_B = 1) \\ (R_{S15})(L_{S15})(L_P)^L(L_B)^B(L_{MI})^{65 \times 62 + 64} (D_B = 0) \end{cases} \quad (13)$$

$$K_N = \begin{cases} (R_{S15})(L_{S15})(L_P)^L(L_B)^B X_{MA}(L_{MI})^{65 \times 62 + 64} (D_B = 1) \\ (R_{S15})(L_{S15})(L_P)^L(L_B)^B X_{MA}(L_{MI})^{65 \times 62 + 63} (D_B = 0) \end{cases} \quad (14)$$

Table 11 Code words for encoding techniques

Code words for PCTM5B technique			
Data Word	Code Word	Data Word	Code Word
000	00000	1000	01000
0001	00001	1001	01001
0010	00010	1010	01010
0011	10101	1011	10100
0100	00100	1100	01100
0101	00101	1101	10010
0110	00110	1110	10001
0111	10110	1111	10000

Code words for PCTM6B technique			
Data Word	Code Word	Data Word	Code Word
0000	000000	1000	001000
0001	000001	1001	001001
0010	000010	1010	001010
0011	100000	1011	010100
0100	000100	1100	100010
0101	000101	1101	010010
0110	010101	1110	010001
0111	100001	1111	010000

4.4.2. PCTM6B ENCODING TECHNIQUE

The codes used in this 6-bit encoding technique are shown in Table 11. This encoding technique requires 32 additional bits for a 64-bit data word, and increases the number of MRs by 50%. To facilitate simultaneous transfer of an entire packet in Corona, which requires 512-bits before encoding, we increase DWDM-degree in MWSR-waveguides from 64 to 66 and increase MWSR-waveguides in each channel from 4 to 6. To distribute optical power between these waveguides, there is also a need to replace 1×4 splitters with 1×6 splitters. The modified versions of equations (10), (11) for worst-case signal and crosstalk-noise power in Corona are shown below:

$$K_S = \begin{cases} (R_{S16})(L_{S16})(L_P)^L(L_B)^B (L_{MI})^{66 \times 63} & (D_B = 1) \\ (R_{S16})(L_{S16})(L_P)^L(L_B)^B (L_{MI})^{66 \times 62 + 65} & (D_B = 0) \end{cases} \quad (15)$$

$$K_N = \begin{cases} (R_{S16})(L_{S16})(L_P)^L(L_B)^B X_{MA}(L_{MI})^{66 \times 62 + 65} & (D_B = 1) \\ (R_{S16})(L_{S16})(L_P)^L(L_B)^B X_{MA}(L_{MI})^{66 \times 62 + 64} & (D_B = 0) \end{cases} \quad (16)$$

4.5. EVALUATION STUDIES

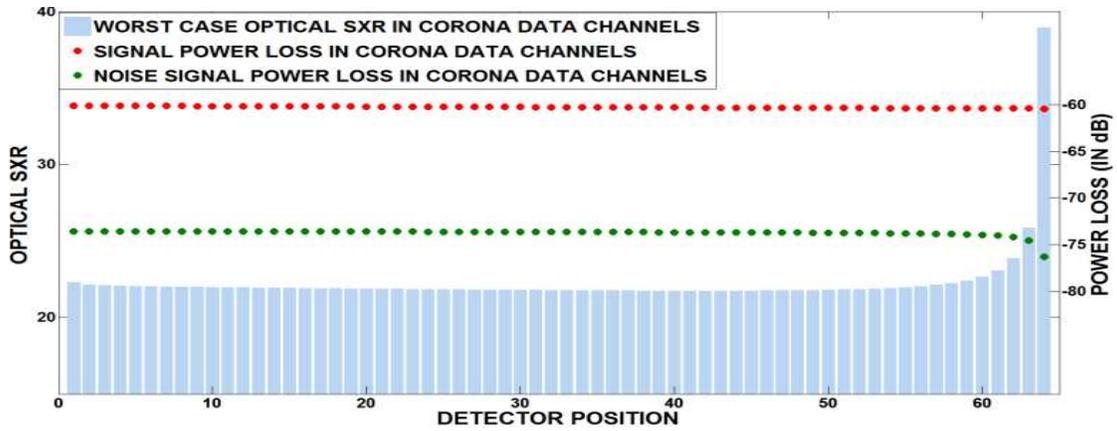
4.5.1. EVALUATION METHODOLOGY

To evaluate the proposed crosstalk-noise mitigation schemes, we implement them on two well-known crossbar-based PNoC architectures: Corona [67] and Firefly [12]. We modeled and simulated our schemes on these architectures using a cycle-accurate NoC simulator. We considered a 256-core single-chip architecture at 22nm for performance analysis. A system-level simulation was performed with the open-source GEM5 architectural simulator with 256 ARM-based cores running parallelized PARSEC benchmarks, to generate traces that were fed into our cycle-accurate NoC simulator. We set a “warm-up” period of 100-million instructions and then captured traces for the subsequent 1-billion instructions. We performed geometric calculations for a 20mm×20mm chip size, to determine lengths of MWSR and SWMR waveguides in the Corona and Firefly PNoCs, respectively. Based on this analysis, we estimated the time for light to travel from the first to the last node as 8-cycles at 5GHz clock frequency in both PNoCs. We used a total packet size of 512-bits as advocated in these architectures, and a DWDM wavelength range in the C and L bands [104], with a starting wavelength of 1530nm.

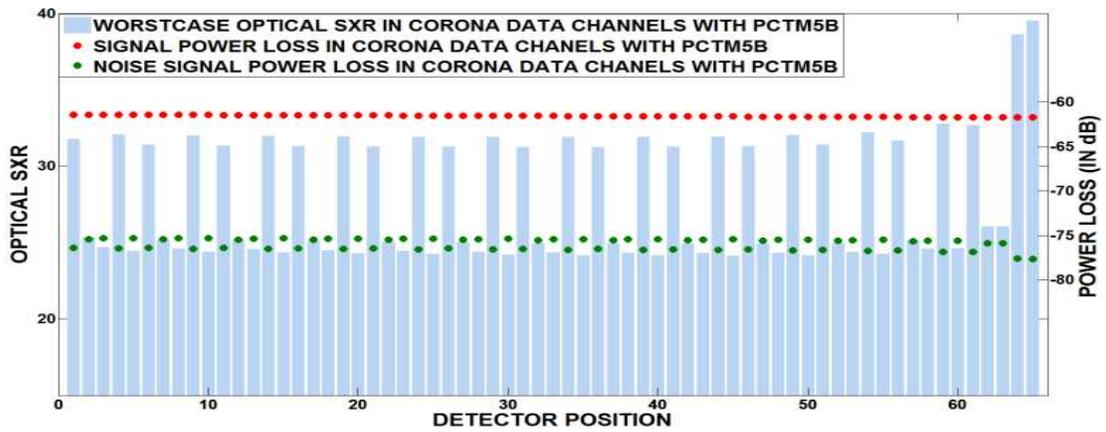
We increased photonic hardware of Corona and Firefly to have a minimal performance (latency, bandwidth) impact and to still enable transfer of an entire encoded-packet from source to destination in one-cycle. PCTM5B (PCTM6B) requires 25% (50%) increase in number of micro-rings and waveguides over the baseline Corona and Firefly architectures. More precisely, PCTM5B has an area overhead of 5.98mm² (electrical) and 8.98mm² (photonic) for Corona; and 11.95mm² (electrical) and 18.69mm² (photonic) for Firefly. PCTM6B has an area overhead of 9.34mm² (electrical) and 17.97mm² (photonic) for Corona; and 18.69mm² (electrical) and 25.34mm² (photonic) for Firefly. These area overheads are reasonably low compared to the much

larger footprint of the chip (400mm^2). The static and dynamic energy consumption of routers and concentrators in the Corona and Firefly PNoCs is calculated with the open-source DSENT tool. Both our schemes also have electrical power overhead: PCTM5B has 0.2W and 0.4W overhead, and PCTM6B has 0.6W and 1.2W overhead for Corona and Firefly, respectively.

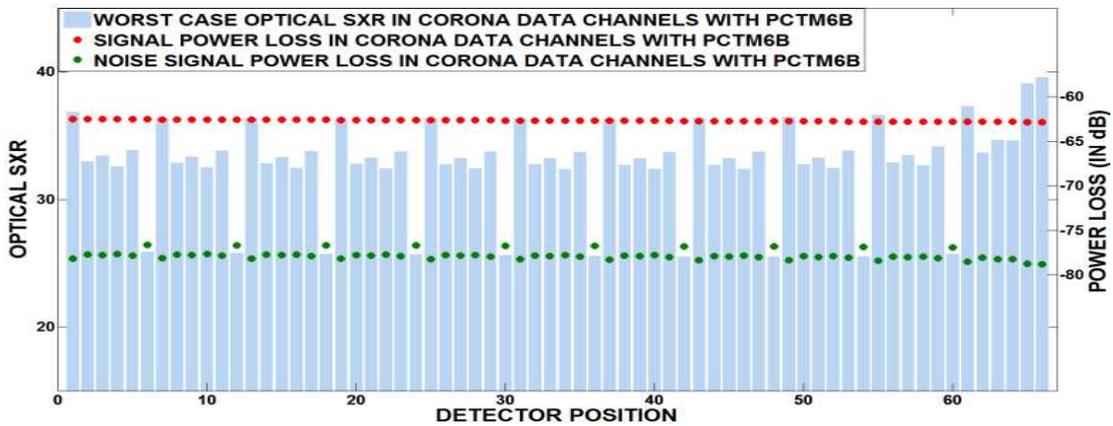
We estimated electrical area and power overhead using gate-level analysis and the open-source CACTI-6.5 tool for memory/buffers. Photonic area overhead is estimated based on the physical dimensions of waveguides, MRs and splitters [104]. For energy consumption of photonic devices, we adopt parameters from [74], [100], with 0.42pJ/bit for every modulation and detection event, and 0.18pJ/bit for driver circuits of modulators and photodetectors. We used photonic-loss values for photonic components, as shown in Table 10, to obtain the photonic laser-power-budget and the corresponding electrical laser power. We consider a one-cycle overhead for both encoding and decoding of data in PCTM5B and PCTM6B, based on our circuit-level analysis at 5GHz. While we consider the baseline token management scheme from [67] for Corona, more sophisticated token management schemes can be employed to further reduce delay overheads of PCTM5B and PCTM6B. Lastly, as our scope of work is limited to optical crosstalk analysis, we consider OSNR as a measure for reliability. A detailed analysis and optimization of the resulting OSNR in the electrical domain at the optical receivers, due to factors such as thermal- and shot-noise is beyond the scope of this work.



(a)



(b)



(c)

Figure 29 Detector-wise signal power-loss, crosstalk-noise power-loss, and minimum optical-OSNR in worst-case power-loss node for Corona (a) baseline with 64-detectors (b) PCTM5B with 65-detectors (c) PCTM6B with 66-detectors.

4.5.2. EVALUATION RESULTS WITH CORONA ARCHITECTURE

Utilizing the models presented in Sections 4.3 and 4.4, we calculate the received crosstalk-noise and OSNR at detectors for the node with worst-case power-loss (N_{WCPL}), which corresponds to detectors in cluster 64 for Corona. We compared the baseline Corona PNoC with fair token-slot arbitration [67] but without any crosstalk-enhancements, with two variants of the architecture corresponding to the two crosstalk-mitigation strategies proposed in this chapter. The worst-case OSNR for the baseline Corona PNoC occurs when all the 64-bits of a received data word in a waveguide are 1's. However, for the implementations of Corona with our crosstalk-mitigation techniques, this is not the case, i.e., each detector in cluster 64 has a worst-case OSNR for a different pattern of 1's and 0's in the received data word. We used our analytical models to determine these unique worst-case patterns for each of the techniques when used with Corona.

Figure 29(a)-(c) show detector signal power-loss, crosstalk-noise power-loss, and OSNR corresponding to the detectors in the 64th cluster for the baseline and two variants of the Corona architecture. Note that the number of detectors in the node (x-axis) varies across the proposed techniques and depends on the number of data bits transmitted in the data waveguide for each technique, as discussed in Section 4.4. Figure 29(b) indicates that worst-case OSNR (lowest value of the bars, which represent OSNR in detectors) improves notably over the baseline shown in Figure 29 (a) when using PCTM5B. However, the improvement is on the lower side for the remaining detectors. Figure 29(c) shows that PCTM6B improves worst-case OSNR marginally over PCTM5B, but does a better job of improving OSNR significantly for most detectors.

From Figure 29, the worst-case OSNR results for the baseline, PCTM5B and PCTM6B techniques are 21.74, 24.13, and 25.50, respectively. The worst-case OSNR is obtained at the 42nd detector of the 64th cluster in the baseline case; whereas for PCTM5B and PCTM6B, worst-case

OSNR occurs at the 45th and 48th detectors of the same cluster, respectively. The worst-case OSNR for Corona from [67] was shown to be close to 14dB – our baseline result (OSNR=21.74) when converted to dB scale (OSNR_{db}=13.4dB) is in line with those results, with a slight difference due to the use of enhanced token-slot arbitration in Corona, compared to [67]. From these results it can be surmised that Corona with PCTM5B and PCTM6B techniques has 11% and 18% improvements in worst-case OSNR compared to the baseline. Both PCTM5B and PCTM6B eliminate occurrences of ‘111’ in a data word and also limit occurrences of ‘11’, to reduce crosstalk-noise in detectors.

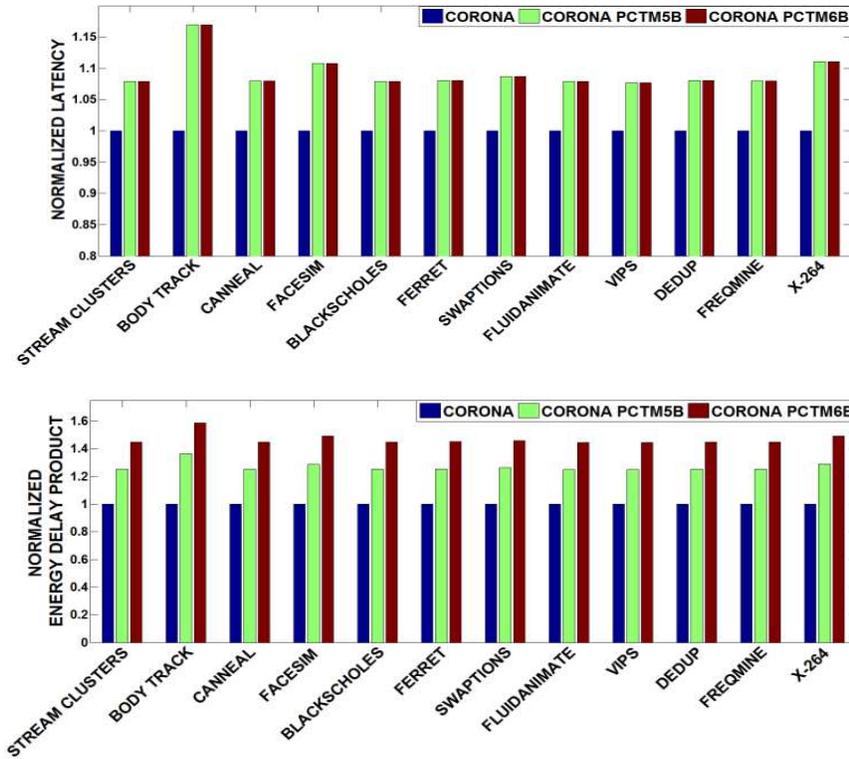
Figure 30(a) shows results that quantify average-packet-latency and energy-delay-product (EDP) for the three Corona configurations, across twelve multi-threaded PARSEC benchmarks. It can be observed that on average, Corona configurations with PCTM5B and PCTM6B have a 9% higher average-latency compared to the baseline. The additional delay due to encoding and decoding of data with PCTM5B and PCTM6B contributes to this latency increase. The Corona configurations with PCTM5B and PCTM6B have 26.5% and 46.2% higher EDP compared to the baseline, respectively. This increase in EDP is not only due to the increase in average latency, but also due to the addition of extra bits for encoding and decoding, which leads to an increase in the amount of photonic hardware in the architectures (more number of MRs, more complex splitters), which increases static energy. Dynamic energy also increases in these architectures, but to a much lesser extent.

4.5.3. EVALUATION RESULTS WITH FIREFLY ARCHITECTURE

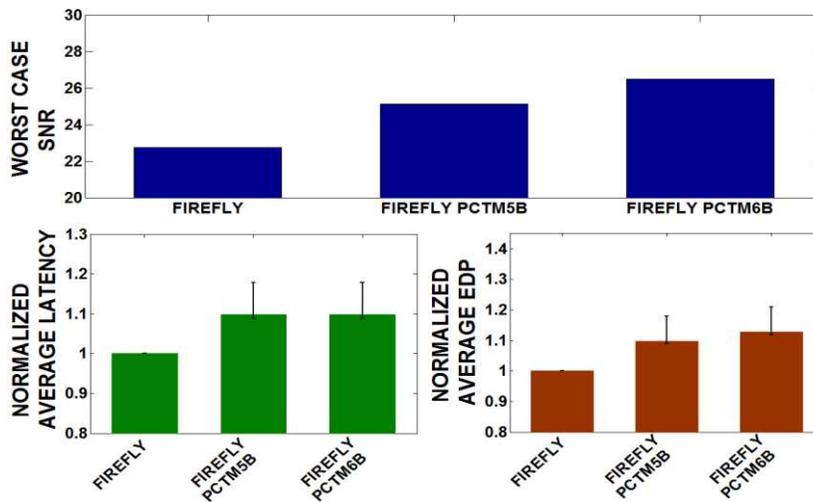
Utilizing static schedule templates for run-time workload management shifts the burden associated with the complex task graph scheduling problem to design-time. However, embedded

systems can encounter unpredictable variations at run-time such as those due to fluctuations in harvested solar energy, slight variations in task execution time on the same core, and randomness of soft error occurrences. To understand how our crosstalk-mitigation techniques behave when ported to a different PNoC architecture, we integrated the techniques with the Firefly [12] crossbar-based PNoC architecture. Unlike the MWSR waveguides used in Corona, Firefly employs reservation-assisted single-write-multiple-reader (R-SWMR) data-waveguides. Each data-channel in Firefly consists of 8 single-write-multiple-read (SWMR) waveguides, with 64-DWDM in each waveguide. Firefly uses only one-eighth of the MRs on each data waveguide compared to Corona, as only eight nodes are capable of accessing each SWMR waveguide. We considered a power waveguide in Firefly similar to that used in Corona and determined that the worst-case-power-loss node (N_{WCPL}) is at the detectors of C_4R_0 , which is the router 0 of cluster 4 in Firefly. Similar to Corona, in Firefly the worst-case signal ($P_{signal(j)}$) and noise power ($P_{noise(j)}$) in the detectors of router C_4R_0 are calculated using Eq.(4)-(9) and OSNR is calculated by Eq.(12). But as Firefly has fewer number of MRs in its data channels, this in turn changes the signal and crosstalk noise power losses before the detector group of N_{WCPL} , so K_S and K_N from Eq.(10) and Eq.(11) are modified to capture the architecture-specific requirements for Firefly.

To implement our PCTM5B and PCTM6B crosstalk-mitigation techniques on Firefly, we propose to make changes similar to that made for Corona, in terms of an appropriate increase in the number of waveguides in each data channel and DWDM in each waveguide, as dictated by the technique. The worst-case OSNR comparison of the baseline Firefly architecture and Firefly configurations with the PCTM5B and PCTM6B techniques is shown in Figure 30 (b) (top-figure). It can be observed that Firefly with the PCTM5B and PCTM6B techniques has 10.5% and 16.5%, improvements in worst-case OSNR over the baseline configuration.



(a)



(b)

Figure 30 (a) Normalized-latency and normalized energy-delay-product (EDP) comparison between Corona baseline and Corona with PCTM5B and PCTM6B, for PARSEC benchmarks. Results are normalized to the baseline Corona results; (b) Worst-case OSNR (on-top), normalized average-latency (bottom-left) and EDP (bottom-right) for PARSEC benchmarks running on the baseline Firefly architecture and Firefly with PCTM5B and PCTM6B.

We also ran simulations with 12 applications from the PARSEC benchmark-suite and obtained normalized average packet-latency and normalized average-EDP for the Firefly configurations, just as we did for the Corona configurations. These results are shown in Figure 30 (b) (bottom-figure), with vertical lines on the bars showing the range of values obtained across the 12 benchmarks. It can be observed that on average, Firefly with PCTM5B and PCTM6B has 9.8% higher average latency compared to the baseline configuration. The PCTM5B and PCTM6B techniques also increase EDP in Firefly by 10% and 12.8% respectively. This increase in latency and EDP of the encoding techniques is due to the additional clock-cycles needed by the encoding and decoding phases as well as the additional bits used by these mechanism, that translate to greater photonic hardware which increases static/dynamic power overheads.

4.5.4. SUMMARY OF RESULTS AND OBSERVATIONS

From the results presented in the previous sections, we can summarize that both of our crosstalk-mitigation techniques significantly reduce crosstalk-noise and improve OSNR in photonic data waveguides. Our techniques have a less than 10% latency overhead (Fig. 3) and less than 5% throughput overhead (results omitted for brevity), on average. The EDP overheads of the techniques are much lower on architectures optimized for physical-layouts such as Firefly, than on non-optimized architectures such as Corona. The PCTM5B technique is a good option to implement on DWDM-based PNoC crossbars with stringent limitations on area and energy overheads and where modest improvements to reliability are sufficient. The PCTM6B technique is a more viable choice for DWDM-based PNoC crossbars that are biased more towards reliability than energy consumption or area concerns, i.e., where higher energy and area overheads are an acceptable price to pay for greater reliability. Finally, while it is possible to extend our work to

create higher order encoding scheme variants with better reliability, e.g., PCTM7B and PCTM8B, we believe that their prohibitively high power and EDP overheads limit their practical applicability.

4.6. CONCLUSIONS

We have presented two crosstalk mitigation techniques for the reduction of crosstalk noise in the detectors of DWDM-based PNoC architectures with crossbar topologies. These techniques (PCTM5B, PCTM6B) show interesting trade-offs between reliability, performance, and energy overhead across two different crossbar-based PNoC architectures. Our experimental analysis on the well-known Corona and Firefly PNoCs has shown that the PCTM5B and PCTM6B techniques can notably improve worst-case OSNR by up to 18%.

5. IMPROVING CROSSTALK RESILIENCE WITH WAVELENGTH SPACING IN PHOTONIC CROSSBAR-BASED NETWORK-ON-CHIP ARCHITECTURES

Crosstalk noise can significantly reduce data transfer reliability in emerging PNoC architectures. Undesirable mode coupling between photonic signals at microring resonators (MR) is the main cause of crosstalk in photonic waveguides. As emerging PNoC architectures employ dense wavelength division multiplexing (DWDM) with multiple cascaded MRs, these architectures suffer from high crosstalk levels. In this chapter, we propose a novel solution to this problem, by increasing the wavelength spacing between adjacent wavelengths in a DWDM waveguide to reduce crosstalk noise. Experimental results on two photonic crossbar architectures (Corona and Firefly) indicate that our approach improves worst-case OSNR by up to 51.7%.

5.1. MOTIVATION AND CONTRIBUTION

A few recent works propose to exploit on-chip photonic links to create PNoC architectures with a crossbar topology [12], [67] that demonstrate improved performance over other topologies. These crossbar-based PNoCs use large numbers of cascaded MRs to support DWDM in their waveguides for parallel data transfers. But crosstalk noise is a major drawback with MRs in these crossbar-based PNoCs, causing severe performance degradation by reducing OSNR in the network. Results in [100] show worst-case OSNR of the Corona crossbar-based PNoC [67] with 64 DWDM in its data channels is close to 14dB. This OSNR value is not sufficient for reliable data communication, as it corresponds to a very high bit error rate (BER), in the order of 10^{-3} .

We observe that for a fixed free spectral range (FSR), increase in DWDM of the waveguide leads to reduction in wavelength spacing between two adjacent wavelengths and this in turn

increases crosstalk noise. From transmission spectrums of cascaded MRs shown in Figure 31, it can be seen that overlapping region between adjacent wavelengths decreases with increase in the wavelength spacing; this in turn reduces crosstalk noise. Thus OSNR in DWDM based photonic crossbars is directly related to the available DWDM in its waveguides. In this chapter, we propose novel wavelength spacing (WSP) techniques to increase spacing between adjacent wavelengths in a DWDM waveguide for PNoCs. Our novel contributions are:

- We propose a novel wavelength spacing (WSP) technique and explore varying levels of WSP to reduce crosstalk noise in DWDM-based crossbar PNoC architectures;
- We explore worst-case OSNR and performance overheads due to WSP on DWDM-based PNoCs such as Corona [12] and Firefly [3] for real-world multi-threaded PARSEC benchmarks.

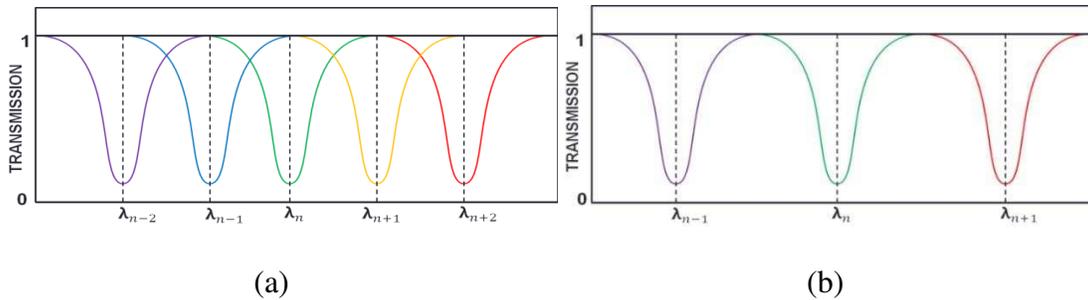


Figure 31 Transmission spectrum of the cascaded microring modulators when using (a) smaller wavelength spacing (b) larger wavelength spacing.

5.2. RELATED WORK

Crosstalk is an intrinsic characteristic of MRs and waveguide crossings. Several prior efforts have analyzed the crosstalk behavior of these components. Crosstalk noise in single waveguide crossings is shown to be close to -47.58 dB [101]. A cascaded MR-based modulator is proposed in [102] for low-density DWDM waveguides, with an extinction ratio of 13dB and negligible

crosstalk. In the aforementioned works, crosstalk noise appears negligible at the device-level. But at the network-level, aggregate crosstalk due to several photonic devices reduces OSNR considerably, creating severe reliability issues. For example, crosstalk analysis of a folded-torus-based PNoC in [103] shows that crosstalk noise power exceeds signal power when network size is equal to or greater than 8×8 nodes. Similar conclusions were drawn from the crosstalk analysis in [100] for the Corona PNoC [67], where its 64 wavelength DWDM data channels are studied and worst-case OSNR is estimated to be 14dB, which is too low for reliable data transmission in practice.

Thus, an emphasis on network-level crosstalk is critical in emerging PNoCs, such as Corona [67] and Firefly [12], otherwise such architectures may not be viable for implementation in future chips. Two encoding techniques PCTM5B and PCTM6B are presented in [28] to improve OSNR in DWDM-based crossbar architectures. Our goal in this chapter is to reduce network-level crosstalk via wavelength spacing (WSP) optimizations. We analyze and quantify the worst-case OSNR, performance, and energy overheads of using variants of our WSP technique with different levels of wavelength spacing in DWDM-based photonic waveguides used in [12], [67].

5.3. WAVELENGTH SPACING (WSP) TECHNIQUE

Microring resonators (MRs) in DWDM-based PNoC architectures can be used as either modulators or detectors. An MR modulator can be operated in two modes: modulating and passing. In the modulating mode, the MR is in resonance with the corresponding resonant wavelength in the waveguide and is capable of removing this wavelength from the waveguide. In the passing mode, the MR simply allows all the wavelengths to pass through undisturbed, as the modulator is out of resonance with all the wavelengths. Similarly, MR detectors can be operated in two modes:

detecting and passing. In the detecting mode, the MR can remove a corresponding resonant wavelength light pulse from the waveguide, whereas in the passing mode it will permit wavelengths to pass through.

Figure 28(a)-(d) show these different modes of operation for an MR modulator and detector. The figures also show crosstalk noise (as dotted/dashed lines) in the modulator and detector MRs during typical modulation and detection modes in the DWDM-waveguide. Whenever a modulator modulates a '0' or a detector detects a '1' from a particular wavelength by removing the light pulse, there is also crosstalk generated in the waveguide, as shown in Figure 28(a) and (d). Thus, MRs generate crosstalk noise, as they not only couple photonic power from their resonance wavelengths but also couple certain portions of photonic power from other wavelengths in the waveguide.

5.3.1. ANALYTICAL MODEL FOR OSNR IN CORONA CROSSBAR-BASED PNO C

Crossbar-based PNOcs such as Corona [67] use cascaded MRs to modulate and detect data from their multiple writer and single reader (MWSR) waveguides. Corona has 64 nodes and each node consists of four processing cores. Inter-node communication is facilitated via a crossbar network with 64 data channels, where each channel has 4 MWSR waveguides with 64 DWDM in each waveguide. This architecture considers a packet size of 512 bits (cache-line size) and is capable of traversing an entire packet from source node to destination node in a single cycle. Note that we also modeled OSNR for the Firefly PNOc [12], but omit its discussion for brevity.

The worst-case OSNR in the Corona crossbar occurs in the detectors of the last (64th) node traversed by the MWSR data channels. This node is called the maximum power loss node (MPLN). Eq.(17) defines OSNR(j) of the jth detector at the MPLN as the ratio of $P_{\text{signal}(j)}$ to $P_{\text{noise}(j)}$ [2]. The

signal power ($P_{\text{signal}(j)}$) and crosstalk noise power ($P_{\text{noise}(j)}$) received at each detector j of MPLN are expressed in Eq. (18) and (19) [100]. $P_S(i,j)$ in Eq. (18) and (19) is the signal power of the i^{th} wavelength received before the j^{th} detector. Similarly in Eq. (20), $P_N(i,j)$ is the crosstalk noise power of the i^{th} wavelength before the j^{th} detector. $\Phi(i,j)$ is the crosstalk coupling factor of the i^{th} wavelength and the j^{th} detector as per Eq. (20). Q refers to the Q-factor of MR and λ_j is resonance wavelength of MR.

$$OSNR(j) = \frac{P_{\text{signal}(j)}}{P_{\text{Noise}(j)}} \quad (17)$$

$$P_{\text{signal}(j)} = L_{DD} P_S(j, j) \quad (18)$$

$$P_{\text{noise}(j)} = L_{DD} P_N(j, j) + \sum_{i=1}^n \Phi(i, j) (P_S(i, j) + P_N(i, j)) \quad (i \neq j) \quad (19)$$

$$\Phi(i, j) = \frac{\delta^2}{\left(\frac{(i-j) \text{FSR}}{N}\right)^2 + \delta^2}, \text{ Here } \delta = \frac{\lambda_j}{2Q} \quad (20)$$

5.3.2. WAVELENGTH SPACING (WSP) TECHNIQUE

Crosstalk noise in an MR depends on the gap between its resonant and non-resonant wavelengths. We observe that the coupling factor (Φ) between these wavelengths increases with a decrease in this gap (Eq. (20)). Therefore, we propose a wavelength spacing (WSP) technique to decrease crosstalk noise at an MR device in DWDM waveguides by increasing spacing between resonant and immediate non-resonant wavelengths. As illustrated in Figure 32, a variable WSP node is added at the beginning of a data waveguide. This node consists of an array of variable-sized MRs capable of switching different spaced wavelengths from a broadband laser source to the data waveguides. To implement the WSP technique in Corona and Firefly, for a fixed FSR, there is a need to decrease DWDM degree in their MWSR and SWMR waveguides. Further due to reduction in DWDM degree, it is not possible to send a packet of 512 bits from source to

destination in one cycle, so the packet size need to be decreased to meet the waveguide DWDM requirements. Depending on the degree of WSP used (see Section 5.4.2), each waveguide is simplified because the number of modulating and detecting MRs is effectively reduced. The reduction in MRs decreases throughput as fewer bits can be transferred in a single cycle. On the other hand fewer MRs also reduce through loss and lower laser power.

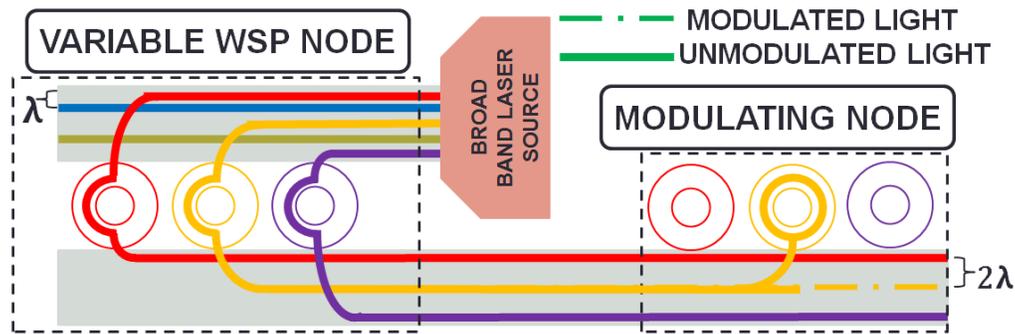


Figure 32 WSP technique: variable WSP-node increases wavelength spacing by 100% from λ to 2λ in the bottom data waveguide of the PNoC and the modulating node on the waveguide modulates on available wavelengths.

5.4. EXPERIMENTS

5.4.1. EXPERIMENTAL SETUP

To evaluate the proposed WSP technique, we implement it on two well-known crossbar PNoC architectures: Corona [67] and Firefly [12]. We modeled and simulated the WSP technique and these PNoCs using a cycle-accurate NoC simulator. We evaluated performance for a 256-core single-chip architecture at a 22nm CMOS node. We used real-world traffic from applications in the PARSEC benchmark suite [43] in our analysis. GEM5 full-system simulation [72] of parallelized PARSEC applications was used to generate traces that were fed into our cycle-accurate NoC simulator. We set a “warm-up” period of 100M instructions and then captured traces for the subsequent 1B instructions. Based on geometric analysis, we estimated the time needed for light

to travel from the first to the last node as 8 cycles at 5 GHz in both architectures, for a 20mm×20mm die. We use a packet size of 512 bits, and a DWDM wavelength range is in the C and L bands [104], with a starting wavelength of 1530nm and FSR of 62 nm. We consider Q-factor (Q) of MR as 9000.

Table 12 Photonic power loss and crosstalk coefficients [100]

Parameter type	Parameter value (in dB)
Propagation loss	-0.274 per cm
Bending loss	-0.005 per 90°
Inactive modulator through loss	-0.0005
Active modulator power loss	-0.6
Passing detector through loss	-0.0005
Detecting detector power loss	-1.6
Active modulator crosstalk coefficient	-16
Detecting detector crosstalk coefficient	-16

Table 13 Worst-case OSNR results for Corona and Firefly architectures

Configuration	Waveguide DWDM	Packet Size (in bits)	Worst-case OSNR
Corona Baseline	64	512	21.74
Corona WSP_20%	53	424	25.39
Corona WSP_40%	46	368	27.91
Corona WSP_60%	40	320	30.13
Corona WSP_80%	36	288	31.6
Corona WSP_100%	32	256	33.04
Firefly Baseline	64	512	22.55
Firefly WSP_20%	53	424	26.22
Firefly WSP_40%	46	368	28.88
Firefly WSP_60%	40	320	31.23
Firefly WSP_80%	36	288	32.82
Firefly WSP_100%	32	256	34.21

The static and dynamic energy consumption of NoC routers and concentrators in Corona and Firefly is based on results from the open-source DSENT tool. We estimated power overhead using

gate-level analysis and CACTI 6.5 for buffers. For energy consumption of photonic devices, we adopt model parameters from recent work [74], [100] with 0.42pJ/bit for every modulation and detection event and 0.18pJ/bit for the driver circuits of modulators and photodetectors. We used photonic loss for photonic components, as shown in Table 12, to determine the photonic laser power budget and correspondingly the electrical laser power.

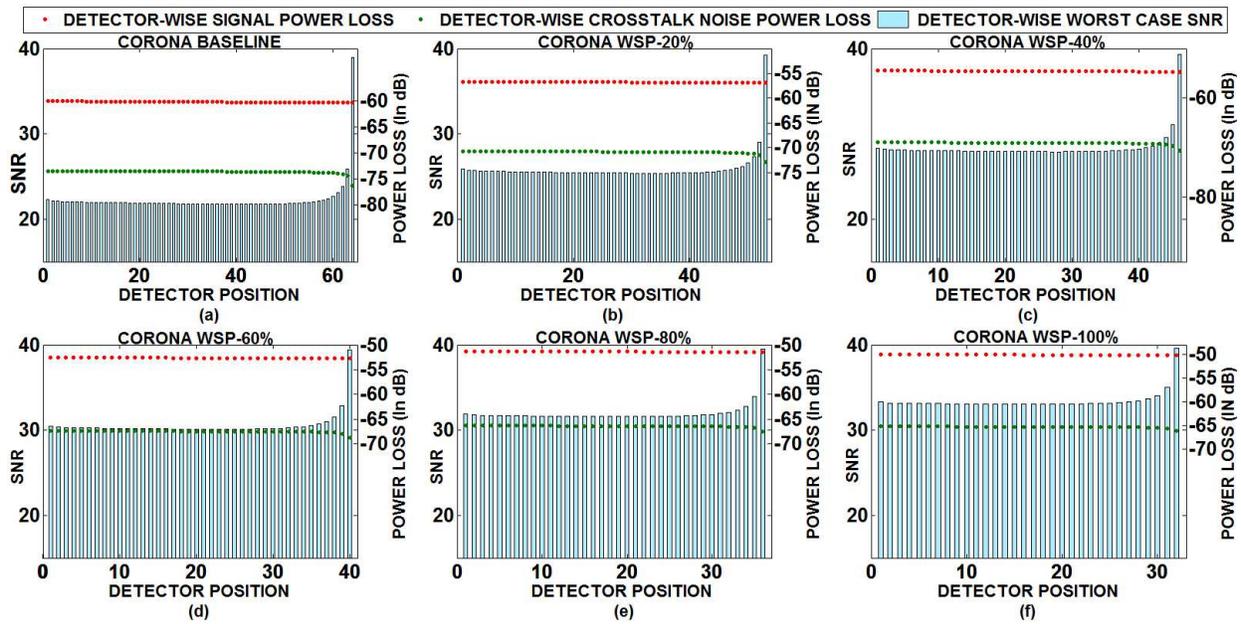


Figure 33 Detector-wise signal power loss, crosstalk noise power loss and minimum OSNR in MPLN for Corona (a) baseline with 64-detectors (b) WSP increased by 20% with 53-detectors (c) WSP increased by 40% with 46-detectors (d) WSP increased by 60% with 40-detectors (e) WSP increased by 80% with 36-detectors (f) WSP increased by 100% (doubled) with 32-detectors.

5.4.2. EXPERIMENTAL RESULTS WITH CORONA AND FIREFLY PNOCS

We compared the baseline Corona PNoC with fair token-slot arbitration [67] and baseline reservation-assisted Firefly PNoC architecture but without any crosstalk-enhancements, with five variants of these architectures corresponding to different degrees of increase in the wavelength spacing: 20% (WSP_20%), 40% (WSP_40%), 60% (WSP_60%) and 100% (WSP_100%). We calculate the received crosstalk noise and photonic OSNR at detectors for the MPLN in Corona,

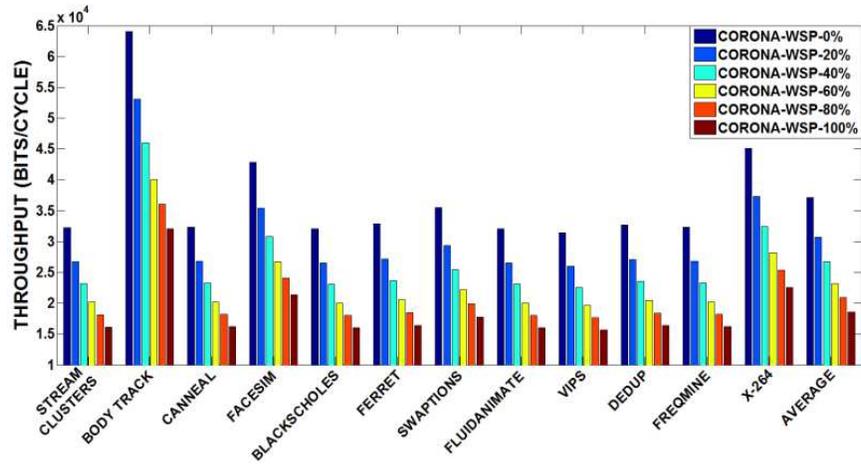
which corresponds to the detectors in node 64, using analytical models presented in Section 5.3. In a similar manner, we also determine the MPLN in Firefly, which is the router 0 of cluster 4. For the Corona and Firefly architectures along with their variants, the worst-case OSNR occurs in the MPLN when all the bits of a received data word in a waveguide are 1's.

Figure 33 (a)-(f) presents detector signal power loss, crosstalk noise power loss, and OSNR corresponding to the detectors in the MPLN for the baseline and five variants of the Corona PNoC. Note that the number of detectors in the node (x-axis) varies across the proposed techniques and depends on the number of data bits transmitted in the data waveguide for each technique, as discussed in Section 5.3. Table 13 summarizes the worst-case OSNR results for all the architectures. The worst-case OSNR in both Corona and Firefly architectures is obtained at the 42nd detector of the MPLN in the baseline case; whereas for the WSP_20%, WSP_40%, WSP_60%, WSP_80% and WSP_100% configurations, worst-case OSNR occurs at the 33th, 27th, 23rd, 20th and 17th detectors of the WPLN, respectively. From the table it can be surmised that Corona with WSP_20%, WSP_40%, WSP_60%, WSP_80% and WSP_100% shows 16.8%, 28.3%, 38.6%, 45.4% and 49.3% improvements in worst-case OSNR compared to the baseline. Furthermore, Firefly with WSP_20%, WSP_40%, WSP_60%, WSP_80% and WSP_100% shows 16.2%, 28%, 38.5%, 45.6% and 51.7% decrease in the worst-case OSNR compared to its baseline. Thus our WSP technique reduces crosstalk and improves OSNR significantly in both Corona and Firefly PNoCs.

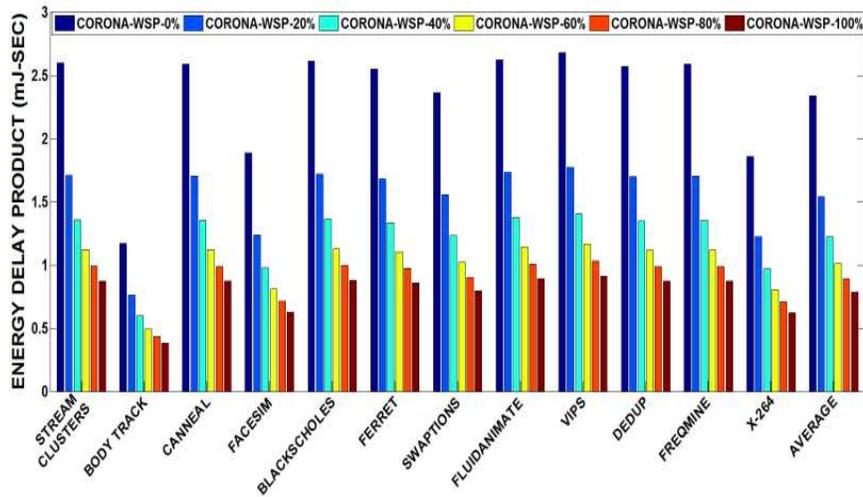
The average throughput and energy-delay product (EDP) for the six configurations of Corona and Firefly architectures are presented in Figure 34 and Figure 35, across 12 multi-threaded PARSEC benchmarks. From Figure 34(a) it can be seen that on average, Corona configurations with WSP_20%, WSP_40%, WSP_60%, WSP_80% and WSP_100% have 17.2%, 28.1%, 37.5%,

43.7% and 50% lower throughput compared to the baseline. Similarly, from Figure 35(a) we observe that on average, Firefly configurations with WSP_20%, WSP_40%, WSP_60%, WSP_80% and WSP_100% have 15.2%, 26.4%, 36.5%, 41% and 48.5% lower throughput compared to the baseline. The decrease in throughput with the WSP technique in these architectures is due to the decrease in number of wavelengths in DWDM for data transfer, as shown in Table 13. In Corona we observe a higher reduction in the throughput compared to Firefly though DWDM was reduced to same extent in both of these architectures. Corona is an all optical crossbar where all data transfers on the chip traverse the optical waveguides (with reduced DWDM), whereas Firefly is a hybrid network where only a certain portion of the on-chip traffic travels through its photonic links (the remaining traffic traverses through its electrical links). Thus, reduction in DWDM has more impact on throughput for Corona compared to Firefly.

From the results for EDP shown in the Figure 34(b), on average Corona configurations with WSP_20%, WSP_40%, WSP_60%, WSP_80% and WSP_100% techniques have 34.1%, 47.8%, 56.8%, 61.8% and 66.36% lower EDP compared to the baseline. From Figure 35(b) we observe that on average, Firefly configurations with WSP_20%, WSP_40%, WSP_60%, WSP_80% and WSP_100% have 17.18%, 25.1%, 35.2%, 40.1% and 49.9% lower EDP compared to the baseline. In general, the WSP technique results in a reduction in energy due to an aggregation of several factors. On the one hand, both Corona and Firefly configurations with WSP_20%, WSP_40%, WSP_60%, WSP_80% and WSP_100% in their data waveguides transmit only 53-bits, 46-bits, 40-bits, 36-bits and 32-bits instead of 64-bits in their respective baselines, which reduces the number of MR modulators and detectors on each waveguide by 17%, 28%, 37.5%, 43.8% and 50% respectively. This reduction in MRs on each waveguide minimizes through loss and decreases laser power, while also minimizing static energy consumption in these architectures.



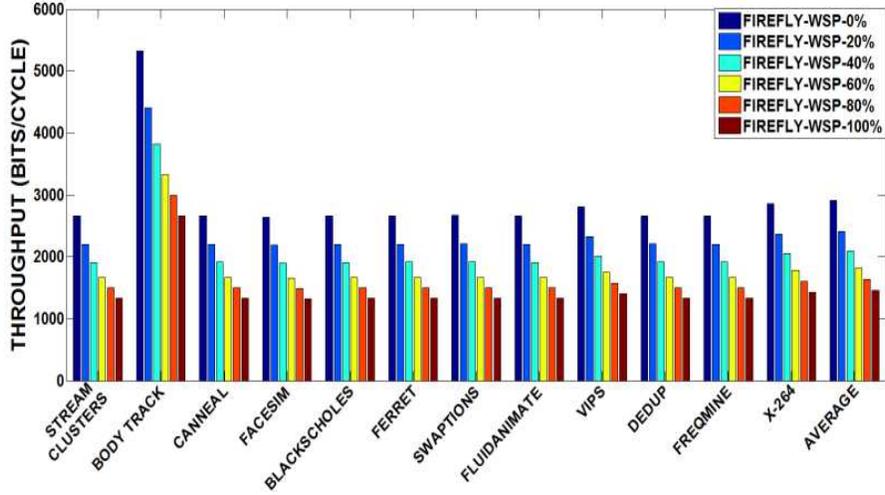
(a)



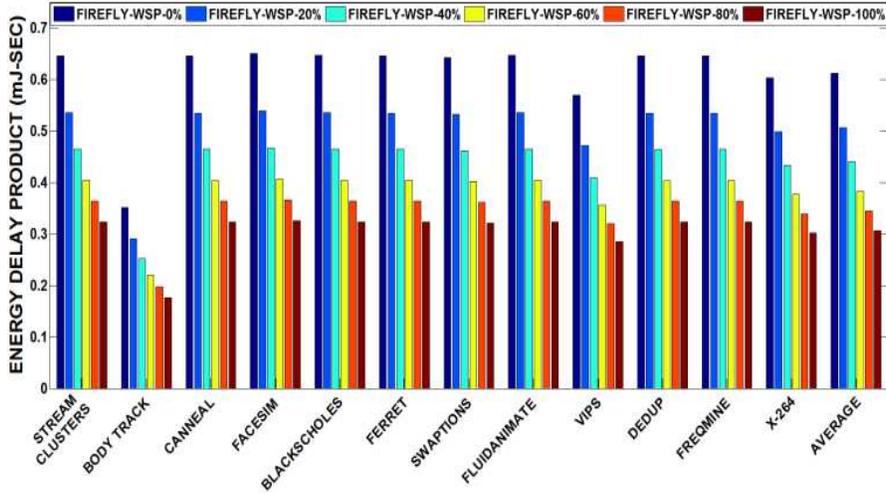
(b)

Figure 34 (a) Throughput, and (b) energy-delay product (EDP) comparison between Corona baseline and Corona configurations with WSP_20%, WSP_40%, WSP_60%, WSP_80% and WSP_100%, for PARSEC suite.

On the other hand, dynamic energy also decreases in all of these configurations compared to its baseline architectures, because fewer bits transverse across data channels, which reduces the energy consumption in modulators, detectors and driver circuits. That is why there is a notable reduction in EDP with the WSP technique.



(a)



(b)

Figure 35 (a) Throughput, and (b) energy-delay product (EDP) comparison between Firefly baseline and Firefly configurations with WSP_20%, WSP_40%, WSP_60%, WSP_80% and WSP_100%, for PARSEC suite.

5.4.3. SUMMARY OF RESULTS AND OBSERVATIONS

From the results presented in the previous subsection, we can summarize that our proposed WSP technique can help to reduce crosstalk noise and improve OSNR in DWDM-based PNoC architectures such as Corona and Firefly. The proposed WSP technique very effectively improves

both reliability and EDP for these architectures. Comparatively higher improvements in EDP were observed in higher through-loss architectures such as Corona compared to architectures with low through-losses such as Firefly.

5.5. CONCLUSIONS

We proposed a WSP technique for the reduction of crosstalk noise in the detectors of dense wavelength division multiplexing (DWDM) based photonic network-on-chip (PNoC) architectures with crossbar topologies. Different WSP configurations (WSP_20%, WSP_40%, WSP_60%, WSP_80% and WSP_100%) of Corona and Firefly show interesting trade-offs between reliability, throughput performance, and energy consumption. Our experimental analysis on the Corona and Firefly PNoCs configurations with WSP_20%, WSP_40%, WSP_60%, WSP_80% and WSP_100% shows improvements in worst-case OSNR by up to 51.7%. This translates into an improvement in bit error rates (BER) in these architectures by up to 100×. Thus the WSP technique can notably improve reliability, with some throughput degradation; however it also reduces EDP due to decrease in photonic hardware and through losses.

6. PICO: MITIGATING HETERODYNE CROSSTALK DUE TO PROCESS VARIATIONS AND INTERMODULATION EFFECTS IN PHOTONIC NOCS

DWDM in photonic links increases susceptibility to intermodulation effects, which reduces OSNR for photonic data transfers. Additionally, process variations induce variations in the width and thickness of MRs causing resonance wavelength shifts, which further reduces OSNR, and creates communication errors. This chapter proposes a novel framework (called *PICO*) for mitigating heterodyne crosstalk due to process variations and intermodulation effects in PNoC architectures. Experimental results indicate that our approach can improve the worst-case OSNR by up to 4.4× and significantly enhance the reliability of DWDM-based PNoC architectures.

6.1. MOTIVATION AND CONTRIBUTION

Prior work indicates that heterodyne crosstalk is a major contributor of crosstalk noise in DWDM-based waveguides, which reduces photonic signal OSNR and reliability in PNoCs [100]. Heterodyne crosstalk noise occurs at a detector MR when it picks up some non-resonant optical power from neighboring wavelengths. The strength of the heterodyne crosstalk noise at a detector MR depends on the following three attributes: (i) channel gap between the MR resonant wavelength and the adjacent wavelengths; (ii) Q-factors of neighboring detector MRs, and (iii) the strengths of the non-resonant signals at the detector. With increase in DWDM, the channel gap between two adjacent wavelengths decreases, which in turn increases heterodyne crosstalk in detector MRs. With decrease in Q-factors of MRs, the widths of the resonant passbands of MRs increases, increasing passband overlap among neighboring MRs, which in turn increases

heterodyne crosstalk. The strengths of the non-resonant signals depend on the losses faced by the non-resonant signals throughout their path from the laser source to the MR detector.

Intermodulation (IM) crosstalk has the biggest influence on the last attribute discussed above, causing loss of non-resonant signals in a DWDM waveguide [108]. IM crosstalk occurs when a modulator MR truncates and consequently modulates the passbands of the neighboring non-resonant signals. Thus the level of heterodyne crosstalk and resultant OSNR at the detector depends on the amount of IM passband truncation at the modulator. This motivates mitigating the effects of IM passband truncation on heterodyne crosstalk by controlling the strengths of the non-resonant signals at the detector.

Additionally, fabrication process variations (PV) induce variations in the width and thickness of MRs, which cause resonance wavelength shifts in MRs [21] [22]. PV-induced resonance shifts reduce the channel gap between the resonances of the victim MRs and adjacent MRs, which increases crosstalk and worsens OSNR. The worsening of OSNR deteriorates the bit-error-rate (BER) in a waveguide. For example, a previous study shows that in a DWDM-based photonic interconnect, when PV-induced resonance shift is over 1/3 of the channel gap, BER increases from 10^{-12} to 10^{-6} [23]. Techniques to counteract the PV-induced resonance shifts in MRs involve realigning the resonant wavelengths by using localized trimming [21] or thermal tuning [22]. Localized trimming is the more viable technique as it enables faster and finer grained control that is also not impacted by on-die thermal variations, unlike thermal tuning. However, our analysis has shown that localized trimming increases intrinsic optical signal loss in MRs and waveguides due to the free carrier absorption effect (FCA). This loss decreases Q-factor of MRs, which increases heterodyne crosstalk in MRs and reduces OSNR.

In this chapter, we present a novel crosstalk mitigation framework called *PICO* to enable reliable communication in emerging PNoC-based multicore systems. *PICO* mitigates the effects of IM crosstalk by controlling signal loss of wavelengths in the waveguide and reduces trimming-induced crosstalk by intelligently reducing undesirable data value occurrences in a photonic waveguide based on the PV profile of MRs. Our framework has low overhead and is easily implementable in any existing DWDM-based PNoC without major modifications to the architecture. To the best of our knowledge, this is the first work that attempts to improve OSNR in PNoCs considering both IM effects and PV in its MRs. Our novel contributions are:

- We present device-level analytical models to capture the deleterious effects of localized trimming in MRs. Moreover, we extend this model for system-level heterodyne crosstalk analysis;
- We propose a scheme for IM passband truncation-aware heterodyne crosstalk mitigation (IMCM) to improve worst-case OSNR of MRs by controlling non-resonant signal power;
- We propose a scheme for PV-aware heterodyne crosstalk mitigation (PVCMM) to improve worst-case OSNR of detector MRs by encoding data to avoid undesirable data occurrences;
- We evaluate our proposed *PICO* (PVCMM+ IMCM) framework by implementing it on the well-known Corona crossbar PNoC architecture [67], and compare it with two encoding based heterodyne crosstalk mitigation mechanisms from [28] for real-world multi-threaded PARSEC benchmarks.

6.2. RELATED WORK

Crosstalk noise can be classified as *homodyne* or *heterodyne*. Homodyne crosstalk usually occurs in MRs used as optical injectors, when an injector MR couples optical power of the same

wavelength from two different ports to a single output port. Heterodyne crosstalk occurs in detector and modulator MRs when an MR picks up some optical power from non-resonant signals. As discussed in [100], homodyne crosstalk may either contribute to the noise or cause fluctuation in the signal power, which makes the analysis and mitigation of homodyne crosstalk more complicated and beyond the scope of this work. Thus this chapter focuses on heterodyne crosstalk and propose solutions to mitigate it. In the rest of the chapter, we use the term crosstalk to refer heterodyne crosstalk.

A few prior works have analyzed crosstalk in PNoCs. The effect of crosstalk noise on OSNR is shown to be negligible in WDM systems presented in [102] and [104], as these systems use only four WDM wavelengths per waveguide. In [108], IM effects are shown to be negligible for a WDM link operating at 10 Gb/s. However, in PNoC architectures that use DWDM (e.g., Corona [11] with 64 wavelength DWDM), there exists significant crosstalk noise. The damaging impact of crosstalk noise in the Corona PNoC is presented in [100], where worst-case OSNR is estimated to be 14dB in data waveguides, which is insufficient for reliable data transfers. To mitigate the impact of crosstalk noise in DWDM based PNoCs, two encoding techniques (PCTM5B and PCTM6B) were presented in [28]. In [29] a technique was proposed to increase channel spacing between adjacent DWDM wavelengths, to mitigate crosstalk in MR detectors. However, *none of these works considers the system-level impact of IM effects or PV on crosstalk in DWDM-based PNoCs.*

Fabrication-induced process variations (PV) impact the cross-section, i.e., width and height, of photonic devices such as MRs and waveguides. A few prior works have explored the impact of PV on DWDM-based photonic links at the system-level [23] [105]. In [23], a thermal tuning based approach is presented that adjusts chip temperature using dynamic voltage and frequency scaling (DVFS) to compensate for chip-wide PV-induced resonance shifts in MRs. In [109], a

methodology to salvage network-bandwidth loss due to PV-drifts is proposed, which reorders MRs and trims them to nearby wavelengths. *All of these PV-remedial techniques are network specific and ignore the harmful effects of PV remedies on crosstalk.* Our proposed framework in this chapter is different and novel as it considers the deleterious effects of IM crosstalk and PV-remedial techniques that increase crosstalk noise in detector MRs.

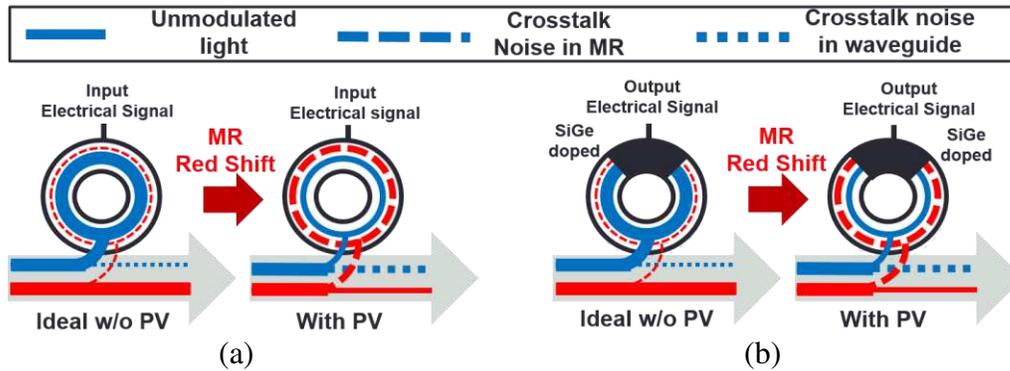


Figure 36 Impact of PV-induced resonance shifts on MR operation in DWDM-based waveguides (note: only PV-induced red resonance shifts are shown): (a) MR as active modulator modulating in resonance wavelength with PV-induced red resonance shifts (b) MR as active detector detecting its resonance wavelength with PV-induced red shifts.

6.3. PV-AWARE CROSSTALK ANALYSIS

6.3.1. IMPACT OF LOCALIZED TRIMMING ON CROSSTALK

An MR can be considered to be a circular photonic waveguide with a small diameter, not to be confused with the larger DWDM-based photonic waveguide for which MRs serve as modulators and detectors. Variations in MR dimensions due to PV cause a “shift” in the resonance wavelengths of MRs. Fig. 1 shows the impact of PV on crosstalk noise (as dotted/dashed lines) in MRs. From Figure 36(a) it can be seen that PV-induced red shifts in MR modulators increase crosstalk noise in the waveguide and decrease signal strength of non-resonating wavelengths. Figure 36(b) shows how PV-induced red shifts increase detected crosstalk noise and decrease

detected signal power of resonance wavelengths in MR detectors, which in turn reduces OSNR and photonic data communication reliability.

As discussed earlier, the localized trimming method is essential to deal with PV-induced resonance red shifts in MRs. However, the use of this method in an MR alters its intrinsic optical properties, which leads to increased crosstalk noise and degraded performance in PNoCs that use these MRs. In this section, we discuss the effects of the localized trimming method on crosstalk and present analytical models to capture these effects in MRs. Further, we extend these models to generate system-level models for the Corona PNoC in order to quantify signal and noise powers in the constituent MRs and DWDM waveguides of the Corona PNoC architecture.

The localized trimming method injects extra free carriers in the circular MR waveguide to counteract the PV-induced resonance red shifts. The introduction of extra free carriers reduces the refractive index of the circular MR waveguide, which in turn induces a blue shift in resonance to counteract the PV-induced red shifts. However, the extra free carriers increase the absorption related optical loss in the MR due to the free carrier absorption effect (FCA) [110]. The increase in the optical loss results in a decrease of MR Q-factor, which increases MR insertion loss and crosstalk, as discussed in Section 6.1.

We use a PV map (described in more detail in Section 6.3.3) to estimate PV-induced shifts in the resonance wavelengths of all the MRs across a chip. Then, for each MR device, we calculate the amount of change in refractive index (Δn_{si}) required to counteract this PV-induced wavelength shift using the following equation [111]:

$$\Delta n_{si} = \frac{\Delta \lambda_r * n_g}{\Gamma * \lambda_r}, \quad (21)$$

where, $\Delta \lambda_r$ is the PV-induced resonance shift that needs to be compensated for, λ_r is the target resonance wavelength of the MR, n_g is the group refractive index (ratio of speed of light to

group velocity of all wavelengths traversing the waveguide) of the MR waveguide, and Γ is the confinement factor describing the overlap of the optical mode with the MR waveguide's silicon core. We assume that the MR waveguides used in this study are similar to those reported in [110], fabricated using standard Si-SiO₂ material with a cross section of 450nm×250nm. The values of Γ and n_g for these MR waveguides are set to 0.7 and 4.2 respectively [110].

The required change in the free carrier concentration to induce the refractive index change of Δn_{si} at around 1.55μm wavelength can be quantified using the following equation [111]:

$$\Delta n_{si} = -8.8 \times 10^{-22} \Delta N_e - 8.5 \times 10^{-18} (\Delta N_h)^{0.8}, \quad (22)$$

where, ΔN_e and ΔN_h are the change in free electron concentration and the change in free hole concentration respectively. The change in the absorption loss coefficient ($\Delta \alpha_{si}$) due to the change in free carrier concentration (owing to the FCA effect) can be quantified using the following equation [111]:

$$\Delta \alpha_{si} = -8.5 \times 10^{-18} \Delta N_e - 6.0 \times 10^{-18} \Delta N_h, \quad (23)$$

The Q-factor of an MR depends on this absorption loss coefficient. The relation between the Q-factor and $\Delta \alpha_{si}$, assuming critical coupling of MRs, is given by the following equation [110], where Q' is the loaded Q-factor of the MR:

$$Q' = Q + \Delta Q = \frac{\pi n_g}{\lambda_r (\alpha + \Delta \alpha_{si})}, \quad (24)$$

where, ΔQ is the change in Q-factor and α is the original loss coefficient, which is a sum of three components: (i) intrinsic loss coefficient due to material loss and surface roughness; (ii) bending loss coefficient, which is a result of the curvature in the MR; and (iii) the absorption effect factor that depends on the original free carrier concentration in the waveguide core. Typically, the localized trimming method injects excess concentration of free carriers into the MR, which increases the absorption loss coefficient (positive $\Delta \alpha_{si}$). As evident from Eq. (24), a positive value

of $\Delta\alpha_{si}$ results in a decrease of the Q-factor. This causes a broadening of the MR passband, which results in increased insertion loss and crosstalk power penalties.

We model the MR transmission spectrum using a Lorentzian function. This function is used to represent coupling factor Φ in Eq. (25) between wavelength λ_i and an MR with resonance wavelength λ_j . From Eq. (23) and (24), it can be inferred that an MR's loaded Q-factor (Q') decreases with localized trimming. This in turn increases Φ and crosstalk noise. Further, using the same function, we determined loss factor γ in Eq. (26) which is the factor by which signal power of a wavelength λ_i is reduced when it passes through an MR whose resonance wavelength is λ_j . Through loss of a wavelength in a waveguide when it passes through an MR is defined as γ times the signal power of all wavelengths received before the MR.

$$\Phi(\lambda_i, \lambda_j, Q') = (1 + (\frac{2Q'(\lambda_i - \lambda_j)}{\lambda_j})^2)^{-1}, \quad (25)$$

$$\gamma(\lambda_i, \lambda_j, Q') = (1 + (\frac{2Q'(\lambda_i - \lambda_j)}{\lambda_j})^{-2})^{-1}, \quad (26)$$

In the next section, we use the derived values of coupling factor Φ and loss factor γ from this section to model worst case crosstalk and OSNR for the Corona PNoC, in the presence of process variations.

6.3.2. PV-AWARE CROSSTALK MODELS FOR CORONA PNO C

We characterize crosstalk in DWDM-waveguides for the well-known Corona PNoC enhanced with token-slot arbitration [67]. In DWDM-based waveguides, data transmission requires modulating light using a group of MR modulators equal to the number of wavelengths supported by DWDM. Similarly, data detection at the receiver requires a group of detector MRs equal to the number of DWDM wavelengths. We present analytical equations to model worst-case

crosstalk noise power, maximum power loss, and OSNR in detector MR groups (similar equations are applicable to modulator MR groups). Before presenting actual equations, we provide notations for the parameters used in the equations, in Table 14 and Table 15.

Table 14 Notations for photonic power loss, crosstalk coefficients [100]

Notation	Parameter type	Parameter value (in dB)
L_P	Propagation loss	-0.274 per cm
L_B	Bending loss	-0.005 per 90°
L_{S12}	1X2 splitter power loss	-0.2
L_{S14}	1X4 splitter power loss	-0.2
L_{S16}	1X6 splitter power loss	-0.2

Table 15 Other model parameter notations

Notation	Crosstalk Coefficient	Parameter Value
Q	Q-factor	9000
L	Photonic path length in cm	
B	Number of bends in photonic path	
λ_j	Resonance wavelength of MR	
R_{S12}	Splitting factor for 1X2 splitter	
R_{S14}	Splitting factor for 1X4 splitter	
R_{S16}	Splitting factor for 1X6 splitter	

The Corona PNoC is designed for a 256-core single-chip platform, where cores are grouped into 64 clusters, with 4 cores in each cluster. A photonic crossbar topology with 64 data channels is used for communication between clusters. Each channel consists of 4 multiple-write-single-read (MWSR) waveguides with 64-wavelength DWDM in each waveguide. As modulation occurs on both positive and negative edges of the clock in Corona, 512 bits (cache-line size) can be modulated and inserted on 4 MWSR waveguides in a single cycle by a sender. A data channel starts at a cluster called ‘home-cluster’, traverses other clusters (where modulators can modulate light and detectors can detect this light), and finally ends at the home-cluster again, at a set of

detectors (optical termination). A power waveguide supplies optical power from an off-chip laser to each of the 64 data channels at its home-cluster, through a series of 1X2 splitters. In each of the 64 home-clusters, optical power is distributed among 4 MWSR waveguides equally using a 1X4 splitter with splitting factor R_{S14} . As all 1X2 splitters are present before the last (64th) channel, this channel suffers the highest signal power loss. Thus, the worst-case signal and crosstalk noise exists in the detector group of the 64th cluster node, and this node is defined as the worst-case power loss node (N_{WCPL}) in the Corona PNoC.

For this N_{WCPL} node, the signal power ($P_{signal}(\lambda_j)$) and crosstalk noise power ($P_{noise}(\lambda_j)$) received at each detector with resonance wavelength λ_j are expressed in Eq. (27) and (28). $P_S(\lambda_i, \lambda_j)$ in Eq. (29) is the signal power of the λ_i wavelength received before the detector with resonance wavelength λ_j . $K(\lambda_i)$ in Eq. (31) represents signal power loss of λ_i before the detector group of N_{WCPL} . $\psi(\lambda_i, \lambda_j)$ in Eq. (30) represents signal power loss of λ_i before the detector with resonance wavelength λ_j within the detector group of N_{WCPL} . Due to PV, crosstalk coupling factor (Φ , Eq. (25)) increases with decrease in loaded Q-factor (Q' , Eq. (24)), which in turn increases crosstalk noise in the detectors. We can define OSNR(λ_j) of the detector having resonance wavelength λ_j of N_{WCPL} as the ratio of $P_{signal}(\lambda_j)$ to $P_{noise}(\lambda_j)$, as shown in Eq. (32).

$$P_{signal}(\lambda_j) = \Phi(\lambda_j, \lambda_j, Q'_{(63 \times 64) + j}) P_S(\lambda_j, \lambda_j), \quad (27)$$

$$P_{noise}(\lambda_j) = \sum_{i=1}^n \Phi(\lambda_i, \lambda_j, Q'_{(63 \times 64) + j}) (P_S(\lambda_i, \lambda_j)) (i \neq j), \quad (28)$$

$$P_S(\lambda_i, \lambda_j) = K(\lambda_i) \psi(\lambda_i, \lambda_j) P_{in}(i), \quad (29)$$

$$\psi(\lambda_i, \lambda_j) = \prod_{k=1}^{(k-1) < j} \gamma(\lambda_i, \lambda_k, Q'_{(63 \times 64) + k}), \quad (30)$$

$$K(\lambda_i) = (R_{S14})(L_{S14})(L_P)^L (L_B)^B \prod_{n=1}^{63} \prod_{j=1}^{64} \gamma(\lambda_i, \lambda_j, Q'_{((n-1) \times 64) + j}), \quad (31)$$

$$OSNR(\lambda_j) = \frac{P_{signal}(\lambda_j)}{P_{Noise}(\lambda_j)}, \quad (32)$$

6.3.3. MODELING PV OF MR DEVICES IN CORONA PNO C

We adapt the VARIUS tool [112] similar to prior work [105] to model die-to-die (D2D) as well as within-die (WID) process variations in MRs. We consider photonic devices with a silicon (Si) core and silicon-dioxide (SiO₂) cladding. VARIUS uses a normal distribution to characterize on-chip D2D and WID process variations. The key parameters are mean (μ), variance (σ^2), and density (α) of a variable that follows the normal distribution. As wavelength variations are approximately linear to dimension variations of MRs, we assume they follow the same distribution. The mean (μ) of wavelength variation of an MR is its nominal resonance wavelength. We consider a DWDM wavelength range in the C and L bands [104], with a starting wavelength of 1550nm and a channel spacing of 0.8nm. Hence, those wavelengths are the means for each MR modeled. The variance (σ^2) of wavelength variation is determined based on laboratory fabrication data [22] and our target die size. We consider a 256-core chip with die size 400 mm² at a 22nm process node. For this die size we consider a WID standard deviation (σ_{WID}) of 0.61nm [105] and D2D standard deviation (σ_{D2D}) of 1.01 nm [105]. We also consider a density (α) of 0.5 [105] for this die size. With these parameters, we use VARIUS to generate 100 process variation maps. Each process variation map contains over one million points indicating the PV-induced resonance shift of MRs. The total number of points picked from these maps equal the number of MRs in the Corona PNoC.

6.4. IM CROSSTALK ANALYSIS

Intermodulation (IM) crosstalk occurs when a resonance wavelength of an MR modulator is modulated by the neighboring MR modulators. As evident from Eq. (26), signal strength of wavelengths in photonic waveguides of DWDM-based PNoCs decrease with increase in loss factor (γ). This γ increases with a decrease in IM gap, which is the gap between resonance wavelengths

of an MR in active and inactive state. Furthermore, this reduction in signal strength of the resonance wavelength also depends on the channel gap (CG) between two adjacent wavelengths in the DWDM. Figure 37 shows the transmission spectrum of MR groups with high and low CG. A change from low DWDM (Figure 37(a)) to higher DWDM (Figure 37(b)) reduces the CG and IM gap, which in turn increases IM crosstalk as is evident from the intersection of the transmission spectrum of inactive MRs with wavelengths in the waveguide ($\lambda_1-\lambda_n$). This IM crosstalk increases wavelength signal loss.

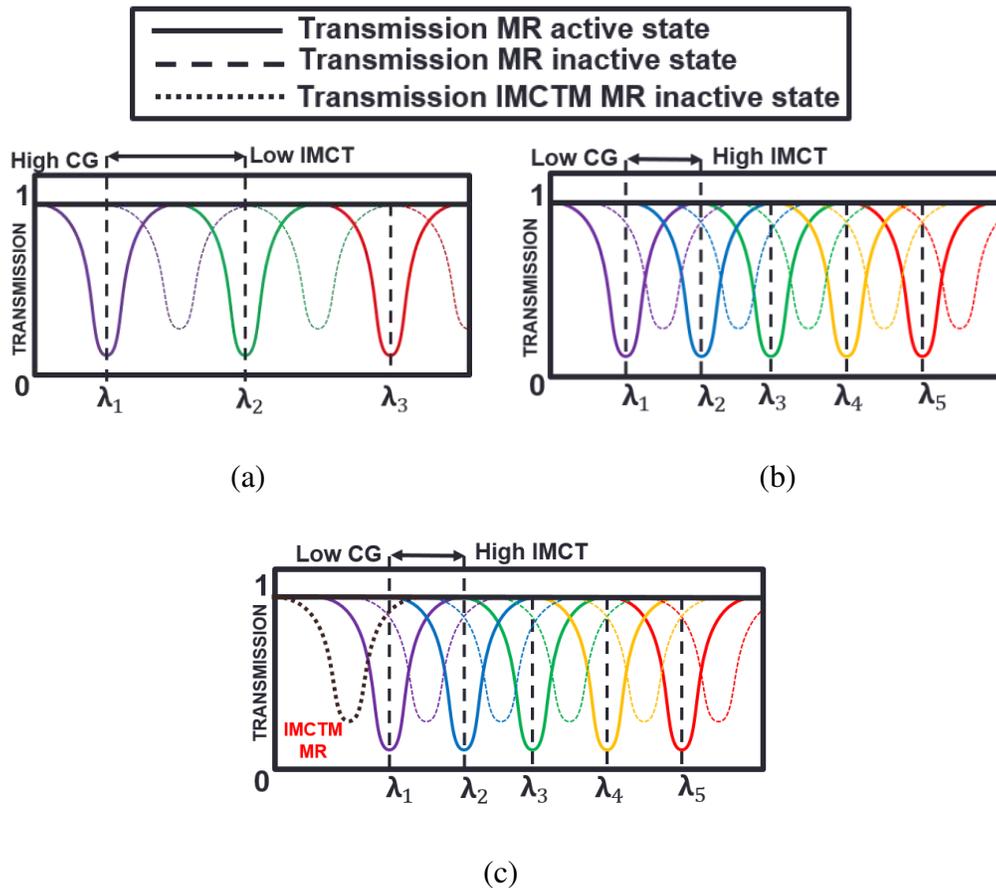


Figure 37 Transmission spectrum of MR groups with (a) high channel gap (CG) (b) low channel gap (CG); (c) IMCTM at low channel gap.

All MR modulators at a node in a DWDM waveguide have neighbors on both sides except the first and the last modulators. So, the first (λ_1) and the last (λ_n) wavelengths of DWDM have the lowest signal losses and highest signal strengths. Thus, the modulated set of DWDM wavelengths that travel along a photonic waveguide to the target detector node have varying signal strengths. At an MR detector group, the first (λ_1) wavelength signal gets filtered and detected by the first detector. As a result, the signal strength of the first wavelength becomes negligible. This negligible signal strength of the first wavelength does not significantly add crosstalk noise in the succeeding neighboring detectors. In contrast, the last (λ_n) wavelength, which also has higher signal strength, gets filtered and detected by the last detector in the detector group. So, the last (λ_n) wavelength signal has to travel along all the detectors in the group of detector rings before being detected. On its way to the last detector, the last wavelength signal incurs crosstalk noise in all the detectors across the detector group. As the strength of the last (λ_n) wavelength signal is high, the incurred crosstalk noise is also high.

6.5. IM-AWARE CROSSTALK MITIGATION

Based on the observations in the previous section, we propose an IM passband truncation aware crosstalk mitigation (IMCM) scheme to decrease crosstalk noise in MRs of DWDM based photonic links. In IMCM, to reduce signal strength of the last wavelength in the DWDM, we propose placing an additional MR at each modulating and detecting node. This extra MR is tuned near to the last (λ_n) wavelength of DWDM with a tuning distance of half the channel gap ($CG/2$) of the DWDM (as shown in Figure 37(c)). This extra MR increases signal loss of this last (λ_n) wavelength and reduces its signal strength. Thus, it creates uniform signal loss across all wavelengths used in the DWDM. This extra MR (passband of this MR is shown with a dotted line

in Figure 37(c)) is always maintained in inactive mode and reduces the effects of IM crosstalk on the boundary wavelengths of DWDM by reducing their respective signal strengths. This mechanism reduces crosstalk in detecting MRs to improve OSNR (and thus reduce BER). To implement the IMCM technique in the Corona PNoC, there is a need to increase the number of MRs in all modulating and detecting nodes by one on their MWSR and SWMR waveguides. The increase in MRs on the waveguides increases through loss and laser power. We account for this overhead in our analysis.

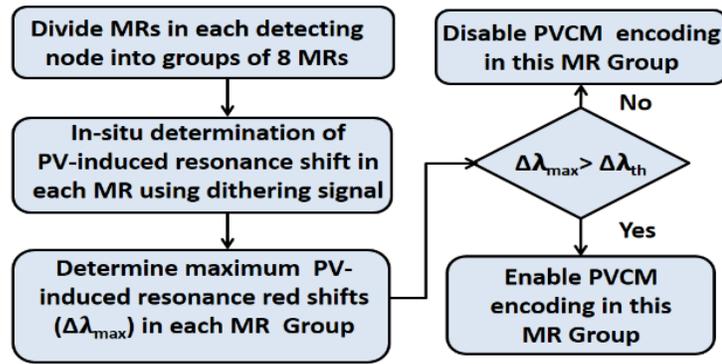


Figure 38 Overview of proposed PVCM technique

6.6. PV-AWARE CROSSTALK MITIGATION

We also propose a PV-aware trimming-induced crosstalk mitigation (PVCM) scheme, which is illustrated in Figure 38. PV-induced red shifts can be realigned using localized trimming, but this process worsens crosstalk noise. From Eq. (25), crosstalk in MR detectors of DWDM-based PNoCs increases with increase in coupling factor (Φ) and increase in signal strength of an immediate non-resonating wavelength. This implies that the trimming-affected crosstalk in a detector can be reduced by reducing the signal strength of immediate non-resonating wavelengths. Therefore, our proposed PVCM technique decreases the signal strength of the immediate non-

resonant wavelength by modulating a zero (shielding bit) on it, which reduces crosstalk noise in the detector. The PVCMM technique first divides detecting MRs into groups of 8 MRs each. Then, it determines the maximum PV-induced resonance red shift ($\Delta\lambda_{\max}$) in each MR group. As discussed in [113], the PV-induced resonance shifts in MRs can be gauged in situ at system initialization by using a dithering signal to generate an anti-symmetric error signal that indicates the magnitude of PV-induced resonance shifts. The overhead of this in-situ PV detection technique can be considered to be negligible [113]. In our analysis, we model and estimate PV in MRs using the VARIUS tool [112], a description of which was given in Section 6.3.3.

Once PV-induced red shifts of MRs are determined, we store information about whether to enable or disable encoding (i.e., injecting shield bits between data bits) for each MR group in a read-only memory (ROM) at the modulating node, based on the maximum PV-induced resonance red shift ($\Delta\lambda_{\max}$) value for the group. If this value is greater than a threshold red shift value ($\Delta\lambda_{\text{th}}$) for an MR group, we store a ‘1’ to enable PVCMM, else we store a ‘0’ to disable PVCMM for this MR group. MR groups with $\Delta\lambda_{\max} < \Delta\lambda_{\text{th}}$ are thus not impacted. Only MR-groups with $\Delta\lambda_{\max} > \Delta\lambda_{\text{th}}$ employ encoding.

6.7. PICO FRAMEWORK: SENSITIVITY ANALYSIS

We combine the IMCM scheme that mitigates the effects of IM crosstalk and the PVCMM scheme that mitigates the PV-affected crosstalk in PNoCs into a holistic crosstalk mitigation framework called *PICO*. As the number of shield bits used in *PICO* increases, laser power and trimming power of PNoCs also increase. Thus, we need to limit the number of shield bits. We performed a sensitivity analysis using the Corona PNoC with varying number of shield bits per detector node to quantify its effect on worst-case OSNR. We analyzed worst case OSNR with 0%,

25%, 50%, 75% and 100% of shield bits added to data bits for the Corona PNoC. Based on our analysis across 100 process variation maps, we determined the value of $\Delta\lambda_{th}$ to be 0.45nm, 0.88 nm, 1.25nm and 4.25nm, for the cases with 25%, 50%, 75% and 100% of shielding bits to data bits, respectively.

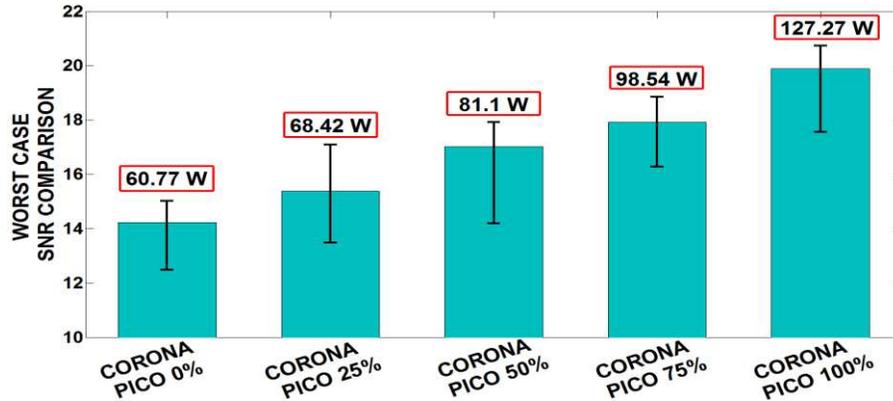


Figure 39 Sensitivity analysis in terms of worst-case OSNR for Corona PNoC with PICO allowing 0%, 25%, 50% and 100% ratio of shield bits to data bits across 100 process variation maps; average power consumption for each configuration is also shown on the top of each bar.

Figure 39 shows the range of worst-case OSNR values across PV maps, for different ratios of shield bits to data bits. From the figure it can be seen that on average *PICO* with 25%, 50%, 75% and 100% shield bits has 8.2%, 19.77%, 26.5% and 40.9% higher worst-case OSNR (note: higher OSNR is better) respectively compared to the baseline (with 0% shielding). Intuitively, higher ratios of shield bits to data bits should result in higher worst case OSNR, as more shield bits can be used to protect data bits, which in turn reduces crosstalk and improves OSNR. But, with increase in number of shield bits, the number of MRs on the waveguides increases, which increases the through losses, requiring more laser power to compensate for the losses. Addressing PV drifts for high MR counts also requires higher trimming power in PNoCs. Figure 39 shows that average power consumption with 25%, 50%, 75% and 100% shield bits is 12.6%, 33.5%, 62.2%

and 109.5% higher compared to the baseline. To balance crosstalk reliability and power overheads, we select the 50% shield bits to data bits configuration for the rest of our experiments.

To implement our *PICO* framework with 50% shielding bits on the Corona PNoC, we increase the number of MWSR waveguides in each channel from 4 to 6, to maintain the same bandwidth as in the baseline case. Additionally, each modulating node needs to store 2,646 bits in its ROM to capture encoding requirements for all the remaining 63 detecting nodes. Power and area overheads for these modifications are presented in the next section. Lastly, we also consider up to a two cycle overhead for encoding and decoding of data in *PICO*. The first cycle is needed to retrieve data from the ROM storage, whereas the second cycle is used only if data is to be encoded before sending on the waveguide.

6.8. EXPERIMENTS

6.8.1. EXPERIMENTAL SETUP

To evaluate our proposed crosstalk noise mitigation framework *PICO* (IMCM+PVCM) in DWDM-based PNoCs, we implement and integrate it with the Corona [67] crossbar-based PNoC. We modeled and performed simulation based analysis of the enhanced Corona PNoC using a cycle-accurate NoC simulator, for a 256 core single-chip architecture at 22nm. As explained in Section 3.3, we generated 100 PV maps to evaluate how *PICO* performs for different PV profiles. We used real-world traffic from applications in the PARSEC benchmark suite [43]. GEM5 full-system simulation [72] of parallelized PARSEC applications was used to generate traces that were fed into our cycle-accurate NoC simulator. We set a “warm-up” period of 100 million instructions and then captured traces for the subsequent 1 billion instructions. We performed geometric calculations for a 20mm×20mm chip size, to determine lengths of MWSR waveguides in the

Corona PNoC. Based on this analysis, we estimated the time needed for light to travel from the first to the last node as 8 cycles at 5 GHz clock frequency. We use a 512 bit packet size, as advocated in the Corona PNoC.

The static and dynamic energy consumption of electrical routers and concentrators in Corona is based on results from the open source DSENT tool. We model and consider area, power, and performance overheads for our framework implemented with the Corona PNoC, as follows. *PICO* has an electrical area overhead estimated to be 6.24 mm² and a power overhead of 1.14 W, using gate-level analysis and the CACTI 6.5 [114] tool for memory and buffers. The photonic area overhead is 9.44 mm², based on the physical dimensions [104] of waveguides, MRs, and splitters. For energy consumption of photonic devices, we adapt model parameters from recent work [73], [74], [100], with 0.42pJ/bit for every modulation and detection event and 0.18pJ/bit for the driver circuits of modulators and photodetectors. We used optical loss for photonic components, as shown in Table 14, to determine the photonic laser power budget and correspondingly the electrical laser power. The MR trimming power is set to 130μW/nm [19] for current injection (blue shift).

6.8.2. EXPERIMENTAL RESULTS WITH CORONA PNOC

Our first set of experiments compares the baseline Corona PNoC with fair token-slot arbitration [67] but without any crosstalk-enhancements, with three variants of the architecture corresponding to the three crosstalk-mitigation strategies we compare: PCTM5B and PCTM6B from [28] and our proposed *PICO* framework from this chapter. PCTM5B and PCTM6B are encoding schemes that replace each 4-bits of a data word with 5-bit and 6-bit code words. These schemes aim to reduce photonic signal-strength of immediate non-resonant wavelengths (adjacent wavelengths in DWDM) to decrease crosstalk and improve OSNR in MR detectors.

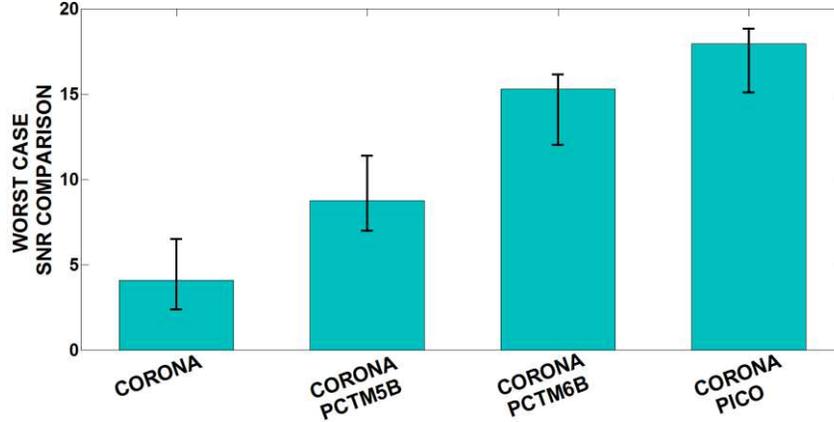


Figure 40 Worst-case OSNR comparison of PICO with PCTM5B [28] and PCTM6B [28] for Corona PNoC considering 100 process variation maps.

Utilizing the models presented in section 6.3, we calculate the received crosstalk noise and OSNR at detectors for the node with worst-case power loss (N_{WCPL}), which corresponds to MR detectors in cluster 64 for the Corona PNoC. While the worst-case OSNR for the baseline Corona PNoC occurs when all of the 64-bits of a received data word in a waveguide are 1's, for the implementations of Corona with PCTM5B, PCTM6B and *PICO*, this is not the case, i.e., each detector in cluster 64 has a worst-case OSNR for a different pattern of 1's and 0's in the received data word. We used our analytical models to determine these unique worst-case patterns for each of the techniques when used with Corona, for an accurate analysis.

Figure 40 summarizes the worst-case OSNR results for the baseline, PCTM5B, PCTM6B, and *PICO*. From the figure, it can be observed that Corona PNoC with *PICO* has 4.4×, 2.05×, and 1.2× OSNR improvements on average, compared to baseline, PCTM5B, and PCTM6B respectively. Both the PCTM5B and PCTM6B techniques eliminate occurrences of '111' in a data word and have limited occurrences of '11', which helps reduce crosstalk noise in the detectors. But these techniques do not consider the impact of IM effects and PV resonance wavelength drifts. More specifically, IM can create significant additional crosstalk with these techniques in some

cases where occurrences of ‘11’ are present. PV in MRs also varies signal power of wavelengths in DWDM as they propagate through the waveguide, so there is need for encoding on specific wavelengths where there is high signal loss (due to trimming) which is not considered in both PCTM5B and PCTM6B. Due to these reasons, PCTM5B and PCTM6B have worse OSNR degradation. *PICO* reduces crosstalk in the detectors by combining benefits from IMCM and using PVCM’s shield bits between data bits. *PICO* also considers the PV profile of MRs to intelligently select MRs for shielding.

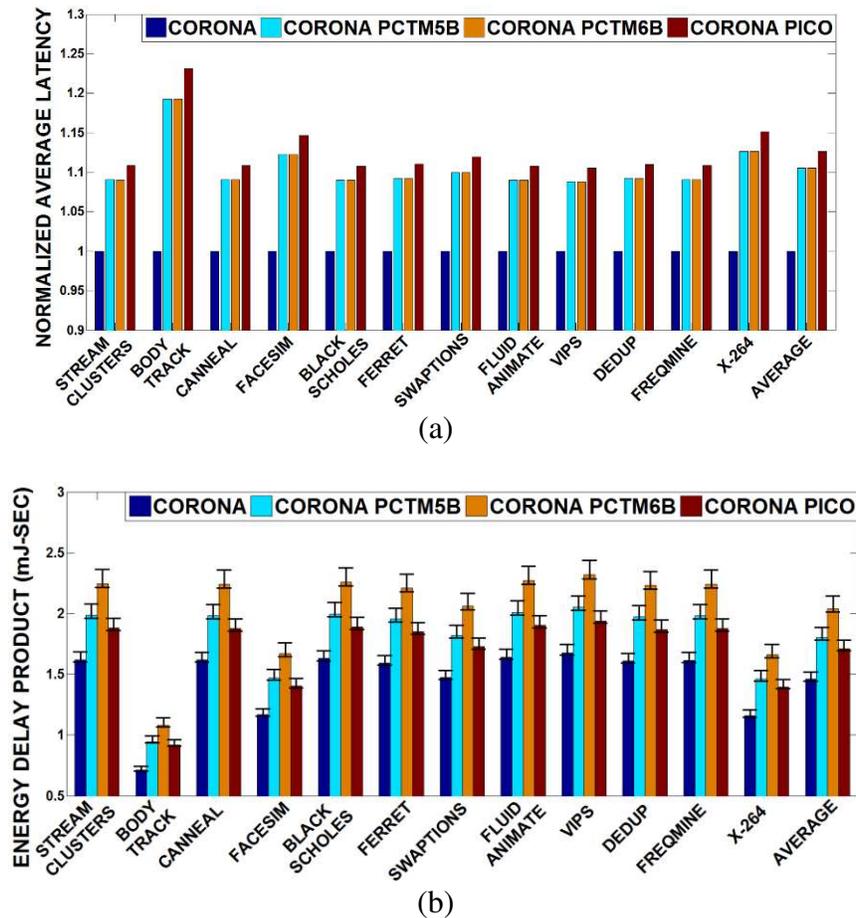


Figure 41 (a) normalized latency and (b) energy-delay product (EDP) comparison between Corona baseline and Corona with PCTM5B, PCTM6B, and *PICO* techniques, for PARSEC benchmarks. Latency results are normalized to the baseline Corona architecture results.

Figure 41 (a) and (b) present detailed simulation results that quantify the average network packet latency and energy-delay product (EDP) for the four Corona configurations. Results are shown for twelve multi-threaded PARSEC benchmarks. From Figure 41(a) it can be seen that on average, Corona with *PICO* has 12.6% higher latency compared to baseline, and it also has 2.1% higher latency compared to both PCTM5B and PCTM6B. The additional delay due to encoding and decoding of data with *PICO*, PCTM5B and PCTM6B contributes to their increase in average latency. The penalty due to encoding/decoding is 1 cycle in PCTM5B and PCTM6B, whereas *PICO* has a 1 or 2 cycle penalty, which increases its delay overhead.

From the results for EDP shown in Figure 41(b), it can be seen that on average, the Corona configuration with our *PICO* framework has 17.2% higher EDP compared to the baseline. Increase in EDP for Corona with *PICO* is not only due to the increase in average latency, but also due to the addition of extra bits for encoding and decoding, which leads to an increase in the amount of photonic hardware in the architectures (more number of MRs, complex splitters). This in turn increases static energy consumption. Dynamic energy also increases in these architectures, but by much less. However, EDP for the *PICO* framework is 5.1% and 16.18% lower compared to PCTM5B and PCTM6B respectively. Despite the higher latency overhead compared to PCTM5B, *PICO* saves considerable dynamic energy compared to PCTM5B as it uses lower number of bits for traversal of the packet. In a similar manner, although *PICO* has higher latency compared to PCTM6B, it conserves laser and trimming/tuning power due to lower photonic hardware requirements than PCTM6B.

6.9. CONCLUSIONS

We have presented a novel heterodyne crosstalk mitigation framework for the reduction of crosstalk noise in the detectors of DWDM based photonic network-on-chip (PNoC) architectures. Our proposed *PICO* framework shows interesting trade-offs between reliability, performance, and energy overhead for the Corona crossbar-based PNoC architecture. Our experimental analysis shows that the *PICO* framework improves worst-case OSNR by 4.4× compared to the baseline Corona PNoC architecture, and by up to 2.05× compared to the best known PNoC crosstalk mitigation schemes from prior work. Thus, *PICO* represents an attractive solution to enhance reliability in emerging DWDM-based PNoCs.

7. HYDRA: HETERODYNE CROSSTALK MITIGATION WITH DOUBLE MICRORING RESONATORS AND DATA ENCODING FOR PHOTONIC NOCS

DWDM in photonic links increases susceptibility to intermodulation and off-resonance filtering effects, which reduces optical signal-to-noise ratio (OSNR) for photonic data transfers. Additionally, process variations induce variations in the width and thickness of MRs causing resonance wavelength shifts, which further reduces OSNR, and creates communication errors. This chapter proposes a novel cross-layer framework called *HYDRA* to mitigate heterodyne crosstalk due to process variations, off-resonance filtering, and intermodulation effects in PNoCs. The framework consists of two device-level mechanisms and a circuit-level mechanism to improve heterodyne crosstalk resilience in PNoCs. Simulation results on three PNoC architectures indicate that *HYDRA* can improve the worst-case OSNR by up to $5.3\times$ and significantly enhance the reliability of DWDM-based PNoC architectures.

7.1. MOTIVATION AND CONTRIBUTION

MRs suffer from intrinsic crosstalk-noise and power-loss due to their design imperfections. Prior work [115] categorizes crosstalk noise into two types: homodyne (coherent) and heterodyne (incoherent). The homodyne crosstalk noise power of a particular wavelength affects the signal power of the same wavelength, whereas with heterodyne crosstalk the signal power gets affected by some noise power of one or more other (different) wavelengths. Heterodyne crosstalk is a major contributor of noise in DWDM-based PNoCs, and reduces OSNR and reliability in PNoCs [115].

Due to the heterodyne crosstalk phenomenon, when a data-modulated wavelength passes by an MR, depending on its data bit-rate (modulation rate), average spectral power, and its relative

detuning from the resonance of the MR, part of its power is dropped by the MR [116]. All modulator, filter, and switch MRs can drop signal power due to heterodyne crosstalk. This heterodyne crosstalk induced signal power drop creates impairments in the passing non-resonant signals. These impairments in a signal result in smoothed transition edges, lengthened rise and fall times, dampened signal amplitude, suppressed signal strength, and reduced extinction ratio, which causes data errors in the signal [117]. The overall impact of these signal impairments is manifested as a power penalty, which is defined as the amount of extra power required at the detector to overcome the data errors caused by these signal impairments.

Heterodyne crosstalk induced signal power drop has an additional effect, referred to as *off-resonance filtering*, at the filter MRs that are coupled with detectors. When a filter MR drops some power from the adjacent non-resonant signals on to a detector at its drop port, this dropped optical power (i.e., crosstalk noise power) produces proportional (pessimistic case) or shot-noise limited (optimistic case) noise current in the detector. This noise current increases the noise floor of the detector, increasing the minimum detectable signal power for the detector. As a result, the detector requires larger signal power to achieve a target OSNR in the presence of this crosstalk noise power. *One of our goals is to reduce crosstalk noise power in detectors due to this off-resonance filtering effect.*

The strength of the heterodyne crosstalk noise power at a detector depends on the following three attributes: (i) channel gap between the MR resonant wavelength and the adjacent wavelength signals; (ii) Q-factors of neighboring detector-coupled filter MRs, and (iii) the strengths of the non-resonant signals at the detector-coupled filter MR. With increase in DWDM, the channel gap between two adjacent wavelength signals decreases, which in turn increases heterodyne crosstalk noise power in detectors. With decrease in Q-factors of MRs, the widths of the resonant passbands

of MRs increase, increasing passband overlap with neighboring non-resonant signals, which in turn increases heterodyne crosstalk noise power. The strengths of the non-resonant signals depend on the losses faced by the non-resonant signals throughout their path from the laser source to the detector-coupled MR filter.

Intermodulation (IM) crosstalk has the biggest influence on the last attribute discussed above, causing suppression (or loss) of signal strength of non-resonant signals in a DWDM waveguide [108]. IM crosstalk occurs when a modulator MR induces impairments in, and as a result, suppresses the neighboring non-resonant signals. Thus the level of heterodyne crosstalk noise power and resultant OSNR at the detector depends on the amount of IM crosstalk induced signal suppression at the modulator. *This motivates mitigating the effects of IM crosstalk induced signal suppression on heterodyne crosstalk by controlling the strengths of the non-resonant signals at the detector.*

Additionally, fabrication process variations (PV) induce variations in the width, thickness, and doping concentration width and thickness of active MRs, which cause resonance wavelength shifts in MRs [22] [30]. PV-induced resonance shifts, when uncompensated, may reduce the gap between the resonances of the victim MRs and adjacent MRs, which increases crosstalk and worsens OSNR. For example, a previous study shows that in a DWDM-based photonic link with 1.48nm channel spacing and 4 Gbps bit-rate, when PV-induced resonance shift is over 1/3rd of the channel gap, bit-error-rate (BER) increases from 10^{-12} to 10^{-6} [23]. Techniques to counteract PV-induced resonance shifts in MRs involve realigning the resonant wavelengths by using localized trimming [18] or thermal tuning [118]. Localized trimming induces a blue shift in the resonance wavelengths (to compensate PV-induced red shifts) of MRs using carrier injection into MRs, whereas thermal tuning induces a red shift in the resonance wavelengths (to compensate PV-

induced blue shifts) of MRs through heating or thermal tuning of MRs using micro-heaters. However, our analysis has shown that localized trimming and thermal tuning increase intrinsic optical loss in MRs and signal loss in waveguides due to the free carrier absorption effect (FCA) [111] and increased optical scattering [119]. *It is important to address this increase in loss, which drives the MR away from critical coupling and decreases its Q-factor, increasing heterodyne crosstalk and reducing OSNR* [31].

In this chapter, we present a novel cross-layer heterodyne crosstalk mitigation framework called *HYDRA* to address the abovementioned challenges and enable reliable communication in emerging PNoC-based manycore chips. Our framework has low overhead and is easily implementable on any existing DWDM-based PNoC without major modifications to the architecture. Our novel contributions are:

- We present device-level analytical models to capture the deleterious effects of localized trimming and thermal tuning in MRs. We also extend these models for system-level heterodyne crosstalk analysis;
- We propose a device-level method for IM effect induced signal suppression aware heterodyne crosstalk mitigation (*IMCM*) that improves worst-case OSNR in detectors by controlling non-resonant signal power;
- We propose another device-level technique for heterodyne crosstalk mitigation (*DMCM*) that uses double MRs to improve worst-case OSNR in detectors by tailoring the MRs' passbands to have steeper roll-off;
- We propose a circuit-level technique for heterodyne crosstalk mitigation (*EDCM*) that improves worst-case OSNR in detectors by encoding data to avoid undesirable data value occurrences;

- We combine *IMCM* (see chapter 6), *DMCM*, and *EDCM* into a holistic cross-layer heterodyne crosstalk mitigation framework called *HYDRA* and evaluate it on three well-known crossbar PNoC architectures as well as prior work on heterodyne crosstalk mitigation.

7.2. RELATED WORK

An important characteristic of photonic signal transmission in on-chip photonic waveguides is that it is inherently *lossy*, i.e., the light signal is subject to losses such as insertion losses in MR modulators and filters [120], propagation and bending loss in waveguides, and splitting loss in splitters. Such losses negatively impact signal strength in waveguides, which reduces OSNR for a given noise power. In addition to the optical signal loss, crosstalk noise of the constituent MRs also deteriorates OSNR. Crosstalk noise in PNoCs usually occurs due to imperfections in MRs used as optical modulators, filters, and switches. This crosstalk noise can be classified as *homodyne* or *heterodyne*.

For homodyne crosstalk, the noise power has the same wavelength as the signal power. As demonstrated in [115], out-of-phase homodyne crosstalk noise always degrades signal integrity. Homodyne crosstalk may either contribute to noise or cause fluctuations in signal power, which makes the analysis and mitigation of homodyne crosstalk complicated and beyond the scope of this work. On the other hand, heterodyne crosstalk occurs when an MR picks up some optical power from non-resonant signals (as explained in Section 7.1). This chapter proposes solutions to mitigate heterodyne crosstalk due to the off-resonance filtering effect. *In the rest of the chapter, we use the term crosstalk to refer to heterodyne crosstalk, unless specified otherwise.*

A few prior works have analyzed crosstalk in PNoCs. The effect of crosstalk noise on OSNR is shown to be negligible in the WDM system presented in [102], as this system uses only four

WDM wavelengths per waveguide with 1.3nm channel spacing and 4 Gbps bit-rate. In [108], IM crosstalk is shown to be negligible for a WDM link operating at 10 Gbps with a channel spacing of 1.6nm. However, in PNoC architectures that use DWDM (e.g., Corona [11] with 64 wavelength DWDM), significant crosstalk noise is expected. The damaging impact of crosstalk noise in the Corona PNoC is presented in [100], where worst-case OSNR is estimated to be 14dB in data waveguides, which is insufficient for reliable data transfers. To mitigate the impact of crosstalk noise in DWDM-based PNoC architectures, two encoding techniques and one wavelength spacing technique were presented in [28], [29]. *However, none of these works considers the system-level impact of IM effects, off-resonance filtering, or process variations on crosstalk noise in DWDM-based PNoCs.*

Fabrication-induced process variations (PV) impact the cross-section, i.e., width and height, of photonic devices such as MRs and waveguides. In MRs, PV causes resonance wavelength drifts, which can be counteracted by using device-level techniques such as localized trimming [18] and thermal tuning [118]. Trimming induces a blue shift in the resonance wavelengths of MRs using carrier injection into MRs, whereas thermal tuning induces a red shift in the resonance wavelengths of MRs through heating of MRs using ring heaters. Such device-level techniques are essential to overcome PV-induced drifts, but they incur high power overheads and may increase signal loss and crosstalk noise, thereby reducing OSNR. This motivates the use of supplementary system-level approaches to reduce the overheads of device-level techniques. A few prior works have explored the impact of PV on DWDM-based PNoCs at the system-level [23], [105]. In [23], a thermal tuning based approach is presented that adjusts chip temperature using dynamic voltage and frequency scaling (DVFS) to compensate for chip-wide PV-induced resonance shifts in MRs. In [105], a methodology to salvage network-bandwidth loss due to PV-drifts is proposed, which

reorders MRs and trims them to nearby wavelengths. *But the achievable benefits for all these supplementary system-level techniques highly depend on the underlying system architecture and they also ignore the harmful effects of device-level PV remedies (i.e., trimming and tuning) on crosstalk.*

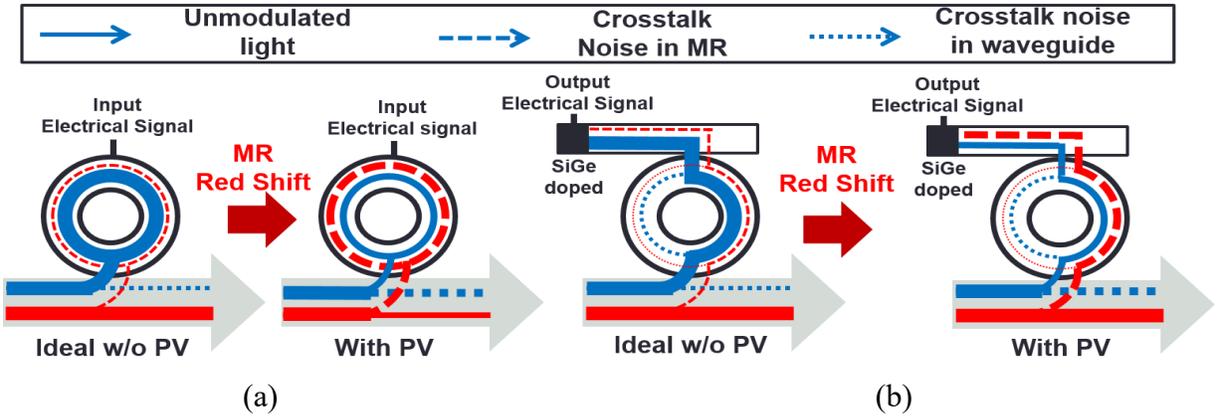


Figure 42 Impact of PV-induced resonance shifts on MR operation in DWDM waveguides (note: only PV-induced red resonance shifts are shown): (a) MR as active modulator with PV-induced red shift, modulating in-resonance wavelength (b) detector-coupled MR filter with PV-induced red shift, filtering its resonance wavelength and dropping it on the detector.

7.3. PV-AWARE CROSSTALK ANALYSIS

An MR can be considered to be a looped photonic waveguide with a small diameter, not to be confused with the straight photonic waveguide used for wavelength-parallel data transfers for which MRs serve as modulators and filters. Variations in an MR’s dimensions due to PV cause a “shift” in its resonance wavelength. Figure 42 shows the impact of PV on crosstalk noise (dashed lines) in MRs. From Figure 42(a), PV-induced red shifts in MR modulators increase crosstalk noise in the waveguide and decrease signal strength of non-resonating wavelength signals. Figure 42(b) shows how PV-induced red shifts increase detected crosstalk noise and decrease detected signal power of resonance wavelengths in detectors, which in turn reduces OSNR and photonic data

communication reliability. As discussed earlier, localized trimming and thermal tuning are essential to deal with PV-induced resonance red and blue shifts in MRs, respectively. However, the use of these methods in an MR alters its intrinsic optical properties, which leads to increased crosstalk and degraded performance in PNoCs that use these MRs.

7.3.1. IMPACT OF LOCALIZED TRIMMING ON CROSSTALK

The localized trimming method injects extra free carriers in the circular MR waveguide to counteract the PV-induced resonance red shift. The introduction of extra free carriers reduces the refractive index of the looped MR waveguide, which in turn induces a blue shift in resonance to counteract the PV-induced red shift. However, the extra free carriers increase the absorption related optical loss in the MR due to the free carrier absorption effect (FCA) [111]. The increase in optical loss results in a decrease of MR Q-factor, which increases MR insertion loss and crosstalk. We use a PV map (described in more detail in Section 7.4) to estimate PV-induced shifts in the resonance wavelengths of all the MRs across a chip. Then, for each MR device, we calculate the amount of change in refractive index (Δn_{si}) required to counteract this PV-induced wavelength shift using the following equation [121]:

$$\Delta\lambda_r = \frac{\Delta n_{eff} * \lambda_r}{n_g} \approx \frac{\Gamma * \Delta n_{si} * \lambda_r}{n_g} \quad (33)$$

where, $\Delta\lambda_r$ is the PV-induced resonance shift that needs to be compensated for, λ_r is the target resonance wavelength of the MR, and n_g is the group refractive index (ratio of speed of light to group velocity of all wavelengths traversing the waveguide) of the MR waveguide. Moreover, Δn_{eff} is the change in effective index that is approximately equal to $\Gamma * \Delta n_{si}$, where Γ is the

confinement factor describing the overlap of the optical mode with the MR waveguide's silicon core. The waveguides used in this study (both MRs' looped waveguides and straight bus waveguides) are rectangular channel waveguides fabricated using Si-SiO₂ material with a cross section of 450nm×220nm. We model these waveguides using a commercial eigenmode solver [122], based on which the values of Γ and n_g at 1550nm are calculated to be 0.78 and 4.16, respectively.

The change in free carrier concentration required to induce refractive index change of Δn_{si} at around 1.55 μ m wavelength can be quantified as follows [111]:

$$\Delta n_{si} = -8.8 \times 10^{-22} \Delta N_e - 8.5 \times 10^{-18} (\Delta N_h)^{0.8}, \quad (34)$$

where, ΔN_e and ΔN_h are the change in free electron concentration and free hole concentration respectively. The change in the absorption loss coefficient ($\Delta \alpha_{si}$) due to the change in free carrier concentration (owing to the FCA effect) can be quantified using the following equation [111]:

$$\Delta \alpha_{si} = 8.5 \times 10^{-18} \Delta N_e + 6.0 \times 10^{-18} \Delta N_h, \quad (35)$$

Quality factor (Q-factor) is a measure of the sharpness of the MR's resonance relative to its central (resonant) wavelength [121]. The Q-factor of MRs affects the magnitudes of crosstalk penalties (as explained in [108] and [123]) and determines the photon-lifetime limited allowable bitrate of signals [38]. Moreover, the Q-factor of an MR represents the number of oscillations of the field in the MR before the circulating field-energy in the MR is depleted to 1/e of the initial energy [121]. Now, from [121], the field-energy decay in the MR cavity depends on the losses in the cavity. Therefore, the Q-factor of an MR depends on the MR's loss coefficient (α) along with

some other parameters. The relationship between the Q-factor and the change in absorption loss coefficient ($\Delta\alpha_{si}$) is given by the Eq. (36) and (37) [121]:

$$Q' = Q + \Delta Q = \frac{2\pi^2 R n_g \sqrt{r_1 r_2 a'}}{\lambda_r (1 - r_1 r_2 a')} \quad (36)$$

$$a' = a + \Delta a = e^{-\pi R (\alpha + \Gamma \Delta\alpha_{si})} \quad (37)$$

where, r_1 and r_2 are the self-coupling coefficients of an add-drop MR (defined in [121]); R is the MR radius; a' is the resultant round-trip field-transmission after an arbitrary change Δa in the original round-trip field-transmission a ; $\Delta\alpha_{si}$ is the change in the MR's original loss coefficient α ; and ΔQ is the change in the loaded Q-factor (Q). Eq. (36) gives the resultant loaded Q-factor Q' for an add-drop MR. Similarly, the Q' for an all-pass MR (described in [121]) can be modeled by setting $r_2=1$ in Eq. (36). Note that, as depicted in Figure 42, we use all-pass MRs as modulators and add-drop MRs as filters and switches.

Now, the original loss coefficient α is a sum of three components: (i) intrinsic loss coefficient due to material loss and sidewall roughness induced scattering loss; (ii) bending loss coefficient, which is a result of the curvature in the MR; and (iii) the absorption effect factor that depends on the original free carrier concentration in the waveguide core. Typically, the localized trimming method (when used to induce a blue-shift in the MR resonance) injects excess concentration of free carriers into the MR, which increases the absorption loss coefficient (positive $\Delta\alpha_{si}$). As evident from Eq. (37), a positive value of $\Delta\alpha_{si}$ results in a decrease in a' , which in turn decreases the Q-factor Q' (from Eq. (36)). This causes a broadening of the MR passband, which results in increased insertion loss, crosstalk noise, and signal impairment/degradation related power penalty.

We model the MR transmission spectrum using a Lorentzian function [124]. In Eq. (38), this function is used to represent coupling factor ϕ [115] between wavelength λ_i and an MR with resonance wavelength λ_j . From [115], we use this coupling factor ϕ to model the heterodyne

crosstalk noise power (of wavelength λ_i) that is dropped on the detector at the drop port of a filter MR with resonance wavelength λ_j . From [108], intermodulation crosstalk incurred by a modulator MR induces signal impairment, suppressing the power in the adjacent signal. As in [108], we use the same Lorentzian function to determine a loss factor γ in Eq. (39), which is the factor by which signal power of a wavelength λ_i is suppressed when it passes by a modulator MR whose resonance wavelength is λ_j . Thus, when a wavelength signal in a waveguide passes by a modulator MR, the intermodulation-crosstalk induced bit-rate independent suppression in its power can be modeled as a through loss, which is defined as γ times the signal power before it passes by the MR.

Now from Eq. (35)-(37), Q' of an MR decreases with localized trimming based increase in carrier concentration. This in turn increases ϕ and crosstalk noise power (Eq. (38)). Note that we do not consider the effect of decrease in free carrier concentration, as we use only carrier injection for both modulation and trimming (to counteract PV-induced red shifts). As would be clear in Section 7.3.2, we do not need to use carrier depletion with trimming, as we would rather heat up the MRs at higher temperatures to counter the PV-induced blue shifts.

$$\Phi(\lambda_i, \lambda_j, Q') = (1 + (\frac{2Q'(\lambda_i - \lambda_j)}{\lambda_j})^2)^{-1}, \quad (38)$$

$$\gamma(\lambda_i, \lambda_j, Q') = (1 + (\frac{2Q'(\lambda_i - \lambda_j)}{\lambda_j})^{-2})^{-1}, \quad (39)$$

7.3.2. IMPACT OF THERMAL TUNING OF MR ON CROSSTALK

As mentioned earlier, localized trimming based carrier injection induces blue shifts in resonance wavelengths of MRs, which can be used to compensate PV-induced red shifts in resonance wavelengths. In contrast, thermal tuning of MRs incurs red shifts in resonance wavelengths of MRs, which can be used to compensate PV-induced blue shifts in resonance wavelengths. From Section 7.3.1, localized trimming results in increased absorption loss

coefficient and subsequent decrease in Q-factor and increase in insertion loss and crosstalk power penalties. Similarly, it can be intuitively inferred that heating of MRs would also increase the absorption loss coefficient in MRs, because, the increase in temperature from the heating of MRs imparts enough energy to some valence electrons of doped silicon (constitutive semiconductor material of MRs) so that they become free carriers. However, these extra free electrons do not significantly increase the net concentration of free carriers in doped silicon. This is because, in doped silicon, the majority of free carriers emanate from the ionization of dopant atoms and usually, all the dopant atoms are completely ionized at room temperature [125]. Thus, any increase in the MR operating temperature above room temperature does not cause ionization of any more dopant atoms. As a result, the concentration of the majority free carriers, and hence, the net free carrier concentration in doped silicon does not change with heating of MRs. Therefore, heating of MRs does not increase the absorption loss coefficient of MRs.

The scattering loss coefficient (that gives fractional loss in signal amplitude) of an MR's circular waveguide is proportional to the refractive index contrast between the core and the cladding ($n_{Si} - n_{SiO_2}$) of the MR waveguide and the size of the surface roughness σ , and is given by the following equation [119] [126]:

$$\alpha_{scatter} = \frac{4(\cos \theta)^3 k_0^2 n_1^2 \sigma^2}{\sin \theta} \cdot \left(\frac{k_0 \sqrt{n_1^2 (\sin \theta)^2 - n_2^2}}{L k_0 \sqrt{n_1^2 (\sin \theta)^2 - n_2^2} + 2} \right) \quad (40)$$

where, $\alpha_{scatter}$ is scattering loss coefficient, k_0 is the free-space wave number, $n_1 = n_{Si}$ is the MR core's refractive index, $n_2 = n_{SiO_2}$ is the MR cladding's refractive index, L is the MR thickness, and θ is the propagation angle for the fundamental mode in the MR. With heating of the MR, the refractive index n_{Si} (of the MR's core) and the refractive index n_{SiO_2} (of the MR's cladding) increase to their new values of $n_{Si} + \Delta n_{Si}$ and $n_{SiO_2} + \Delta n_{SiO_2}$ respectively, which are given by the following equations (41) and (42).

$$n_{Si}^{T+\Delta T} = n_{Si}^T + \Delta n_{Si} = n_{Si}^T + \frac{\delta n_{Si}}{\delta T} \cdot \Delta T, \quad (41)$$

$$n_{SiO_2}^{T+\Delta T} = n_{SiO_2}^T + \frac{\delta n_{SiO_2}}{\delta T} \cdot \Delta T, \quad (42)$$

where, $\delta n_{Si}/\delta T$ and $\delta n_{SiO_2}/\delta T$ are the thermo-optic coefficients of *Si* (MR's core) and *SiO₂* (MR's cladding) materials respectively, and they assume the values of $1.86 \times 10^{-4} \text{ K}^{-1}$ and $1 \times 10^{-5} \text{ K}^{-1}$ respectively [113]. ΔT is an increase in temperature of the MR due to heating. Due to smaller thermo-optic coefficient of *SiO₂* and smaller mode field confinement in *SiO₂* cladding, the effects of temperature change on $n_{SiO_2}^{T+\Delta T}$ is negligible. If a blue shift in an MR's resonance wavelength of $\Delta\lambda_r$ is to be compensated by heating the MR, the required increase in MR's temperature can be computed using the following equation [113].

$$\Delta\lambda_r = \Gamma \cdot \frac{\delta n_{Si}}{\delta T} \cdot \frac{\lambda_r}{n_g} \cdot \Delta T, \quad (43)$$

Now, as the thermo-optic coefficient of *Si* is greater than that of *SiO₂*, $n_{Si}^{T+\Delta T}$ increases faster with increase in temperature than $n_{SiO_2}^{T+\Delta T}$. As a result, the difference $(n_1^2(\sin\theta)^2 - n_2^2)$ in Eq. (40), which depends on the index contrast between the core and the cladding, increases with increase in temperature. This leads to an increase in $\alpha_{scatter}$ with increase in temperature (Eq. (40)). Now, similar to the case of localized trimming, this increase in scattering loss coefficient leads to decrease in MR Q-factor. Using Eq. (40), the increased value of scattering loss coefficient $\alpha_{scatter}$ can be calculated, which then can be used in place of $(\alpha + \Delta\alpha_{Si})$ in Eq. (37) to find the decreased value of Q-factor from Eq. (36).

To model and compare the effects of localized trimming and thermal tuning of MRs, we simulate an MR with a radius (*R*) of $1.8 \mu\text{m}$ (deemed as implementable with CMOS-type processes based on projections from [127]) considering initial original Q-factor of 12500, self-coupling coefficients $r_1=0.99$, $r_2=0.99$, and field-transmission coefficient *a* of 0.991. Note that we use the

initial $Q=12500$, because it gives the optimum value of total MR filter penalty for 5Gbps bitrate and 64 channels (as projected from [41]). Also, note that we assume initial $\alpha_{scatter}=0.14\text{cm}^{-1}$, which corresponds to $\sigma=1\text{nm}$, $n_{Si}=3.5$, $n_{SiO_2}=1.5$, $L = 220\text{nm}$, and $\theta=26.51$ in Eq. (40).

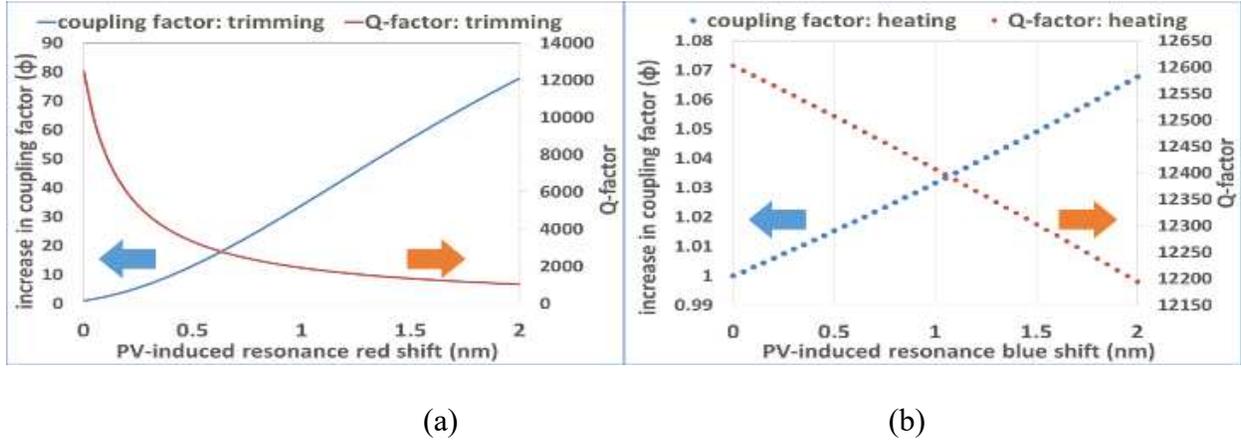


Figure 43 (a) Effect of localized trimming, (b) effect of thermal tuning, on the Q-factor and fractional increase in coupling factor of an example MR. Here, the fractional increase in coupling factor is calculated w.r.t. the original coupling factor of the MR without PV.

Using Eq. (33)-(42), we evaluate the values of Q-factor and increase in coupling factor ϕ for this example MR, when PV-induced red/blue shifts of different values in the resonance wavelength of this MR are compensated by using localized trimming/thermal tuning. Figure 43(a)-(b) plot these values of Q-factor and ϕ for localized trimming and thermal tuning respectively. From the figure, compensating 2nm PV-induced red shift in an MR's resonance wavelength with localized trimming decreases the MR's Q-factor by 91.7% and increases ϕ by $77.8\times$ compared to original Q-factor and coupling factor, respectively. Furthermore, compensating 2nm of PV-induced blue shift in MR's resonance wavelength with thermal tuning decreases the MR's Q-factor by only 3.25% and increases ϕ by $1.07\times$ compared to the original Q-factor and coupling factor, respectively. Thus, it can be concluded that thermal tuning of MRs has negligible impact on MRs' Q-factor and

coupling factor compared to localized trimming. Therefore, compared to localized trimming, thermal tuning does not significantly increase insertion loss and crosstalk penalties for MRs.

However, note that thermal tuning cannot compensate for PV-induced red shifts in MRs' resonance wavelengths. Therefore, in a typical PNoC, where both red and blue shifts in MRs' resonance wavelengths are present, the use of localized trimming is inevitable. As a result, it is imperative to overcome the poor efficiency of localized trimming. We propose, as part of our *HYDRA* framework, a circuit-level data encoding technique (*EDCM*; Section 7.6) that mitigates the effect of PV-remedial techniques (both localized trimming and thermal tuning) on MR crosstalk penalties. Furthermore, this chapter only analyzes the impact of PV and its remedial techniques on crosstalk noise. Evaluating the impact of thermal variations on crosstalk noise is beyond the scope of this chapter. In the next subsection, we use the derived values of Φ and γ from this and the previous section to model worst-case crosstalk and OSNR for the Corona PNoC, in the presence of process variations.

7.3.3. PV-AWARE CROSSTALK MODELS FOR CORONA PNOC

We characterize crosstalk in waveguides with DWDM for the Corona PNoC enhanced with token-slot arbitration [67]. We present equations to model the off-resonance filtering effect induced crosstalk noise power and resultant OSNR in the detectors of receiver groups. Before presenting actual equations, we show notations for parameters used in the equations, in Table 16 and Table 17.

The Corona PNoC is designed for a 256-core single-chip platform, where cores are grouped into 64 clusters, with 4 cores in each cluster. A photonic crossbar topology with 64 data waveguide groups is used for communication between clusters. Each data waveguide group consists of 4

multiple-write-single-read (MWSR) waveguides with 64-wavelength DWDM in each waveguide. As modulation occurs on both positive and negative edges of the clock in Corona, 512 bits (cache-line size) can be modulated and inserted on 4 MWSR waveguides in a single cycle by a sender. Each of the 64 data waveguide groups starts at a different cluster called ‘home-cluster’, traverses other clusters (where modulators can modulate light and receivers can filter and detect this light), and finally ends at the home-cluster again, at a set of receivers (optical termination).

Table 16 Photonic power loss, crosstalk coefficients [74], [100]

Notation	Parameter type	Parameter value (in dB)
L_P	Propagation loss	-0.274 per cm
L_B	Bending loss	-0.0085 per 90°
L_{S12}	1×2 splitter power loss	-0.2
L_{S14}	1×4 splitter power loss	-0.2
L_{S16}	1×6 splitter power loss	-0.2

Table 17 Other model parameter notations [74]

Notation	Crosstalk Coefficient	Parameter Value
Q	Q-factor	9000
RS	Detector responsivity	0.8 A/W
L	Photonic path length in cm	
B	Number of bends in photonic path	
λ_j	Resonance wavelength of MR	
R_{S12}	Splitting factor for 1×2 splitter	
R_{S14}	Splitting factor for 1×4 splitter	
R_{S16}	Splitting factor for 1×6 splitter	

A power waveguide supplies optical power from an off-chip laser to each of the 64 data waveguide groups at its home-cluster via a series of 1×2 splitters. In each of the 64 home-clusters, optical power is distributed among 4 MWSR waveguides equally using a 1×4 splitter with splitting factor R_{S14} . As all 1×2 splitters are present before the last (64th) waveguide group, this waveguide group suffers the highest signal power loss. Therefore, the worst-case signal and crosstalk noise

exists in the detectors of the receiver group of the 64th cluster node, and this node is called the worst-case power loss node (N_{WCPL}) in the Corona PNoC.

For this N_{WCPL} node of the Corona PNoC, the signal power ($P_{signal}(\lambda_j)$) and crosstalk noise power ($P_{noise}(\lambda_j)$) received at a receiver (i.e., detector-coupled MR filter) with resonance wavelength λ_j are expressed in Eq. (44) and (45) respectively. $K(\lambda_i)$ in Eq. (46) represents signal power loss of λ_i before the receiver group of N_{WCPL} . $\psi(\lambda_i, \lambda_j)$ in Eq. (47) represents signal power loss of λ_i before the receiver with resonance wavelength λ_j within the receiver group of N_{WCPL} . $P_S(\lambda_i, \lambda_j)$ in Eq. (48) is the signal power of the λ_i wavelength in the waveguide that has reached the receiver with λ_j resonance wavelength in the receiver group of N_{WCPL} after passing through all the preceding receivers. Due to PV (more details about modeling of PV in PNoCs are presented in the next subsection), crosstalk coupling factor (ϕ , Eq. (38)) increases with decrease in loaded Q-factor (Q' , which is calculated by using Eq. (36) and Eq. (37)), which in turn increases off-resonance filtering effect induced crosstalk noise in the detectors. Furthermore, $Q'_{(x \times y) + j}$ is defined as the Q-factor of j^{th} MR which is in the $x+1^{\text{th}}$ node and each node is having 'y' number of MRs. We can define OSNR(λ_j) at the detector in the receiver (with resonance wavelength λ_j) of N_{WCPL} as the ratio of $P_{signal}(\lambda_j)$ to $P_{noise}(\lambda_j)$, as shown in Eq. (49). These equations (i.e., (44)-(49)) are based on the models presented in the prior works [100] and [115].

$$P_{signal}(\lambda_j) = \Phi(\lambda_j, \lambda_j, Q'_{(63 \times 64) + j}) P_S(\lambda_j, \lambda_j), \quad (44)$$

$$P_{noise}(\lambda_j) = \sum_{i=1}^n \Phi(\lambda_i, \lambda_j, Q'_{(63 \times 64) + j}) (P_S(\lambda_i, \lambda_j)) (i \neq j), \quad (45)$$

$$K(\lambda_i) = (R_{S14})(L_{S14})(L_P)^L (L_B)^B \prod_{n=1}^{63} \prod_{j=1}^{64} \gamma(\lambda_i, \lambda_j, Q'_{((n-1) \times 64) + j}), \quad (46)$$

$$\psi(\lambda_i, \lambda_j) = \prod_{k=1}^{(k-1) < j} \gamma(\lambda_i, \lambda_k, Q'_{(63 \times 64) + k}), \quad (47)$$

$$P_S(\lambda_i, \lambda_j) = K(\lambda_i) \psi(\lambda_i, \lambda_j) P_{in}(i), \quad (48)$$

$$OSNR(\lambda_j) = \frac{P_{signal}(\lambda_j)}{P_{Noise}(\lambda_j)}, \quad (49)$$

7.3.4. MODELING PV OF MR DEVICES IN CORONA PNO C

We adapt the VARIUS tool [112], similar to prior work [105], to model die-to-die (D2D) as well as within-die (WID) process variations in MRs. We consider photonic devices with a silicon (Si) core and silicon-dioxide (SiO₂) cladding. VARIUS uses a normal distribution to characterize on-chip D2D and WID process variations.

The key parameters are mean (μ), variance (σ^2), and density (ω) of a variable that follows the normal distribution. As wavelength variations are approximately linear to dimension variations of MRs, we assume they follow the same distribution. The mean (μ) of wavelength variation of an MR is its nominal resonance wavelength. We consider a DWDM wavelength range in the C and L bands [104], with a starting wavelength of 1550nm and a channel spacing of 0.8nm. Hence, those wavelengths are the means for each MR modeled. The variance (σ^2) of wavelength variation is determined based on laboratory fabrication data [22] and our target die size. We consider a 256-core chip with die size 400 mm² at a 22nm process node. For this die size we consider a WID standard deviation (σ_{WID}) of 0.61nm [105] and D2D standard deviation (σ_{D2D}) of 1.01nm [105]. We also consider a density (ω) of 0.5 [105] for this die size, which is the parameter that determines the range of WID spatial correlation required by the VARIUS tool. With these parameters, we use VARIUS to generate 100 PV maps, these maps are used to model PV in Corona PNoC.

7.4. HYDRA FRAMEWORK: OVERVIEW

Our proposed cross-layer *HYDRA* framework enables crosstalk resilience in DWDM-based PNoC architectures by integrating device-level and circuit-level enhancements that seamlessly

work together. Figure 44 gives a high-level overview of our framework. The IM effects induced signal suppression aware crosstalk mitigation (*IMCM*) scheme employs additional MRs to decrease wavelength-specific crosstalk noise at the detectors of DWDM-based photonic links. For more details of *IMCM* scheme refer to Section 6.5 of chapter 6. The double MR based crosstalk mitigation mechanism (*DMCM*) employs double microrings (DMRs) as signal filters to reduce the crosstalk noise at the detectors. This technique improves OSNR in DWDM-based photonic links. However, excessive usage of DMRs (or higher-order filters) increases area, PV redress power (static power required to counter PV-induced resonance drifts in the DMRs) and laser power overheads for PNoC architectures [128]. Thus, to reduce these overheads, we also devise a circuit-level crosstalk mitigation mechanism (*EDCM*) that uses a 5-bit encoding mechanism to intelligently reduce undesirable data value occurrences in a photonic waveguide. This allows for further reduction in crosstalk noise and more effectively improves OSNR in DWDM-based PNoC architectures. The next three sections present details of the *IMCM*, *DMCM*, and *EDCM* techniques.

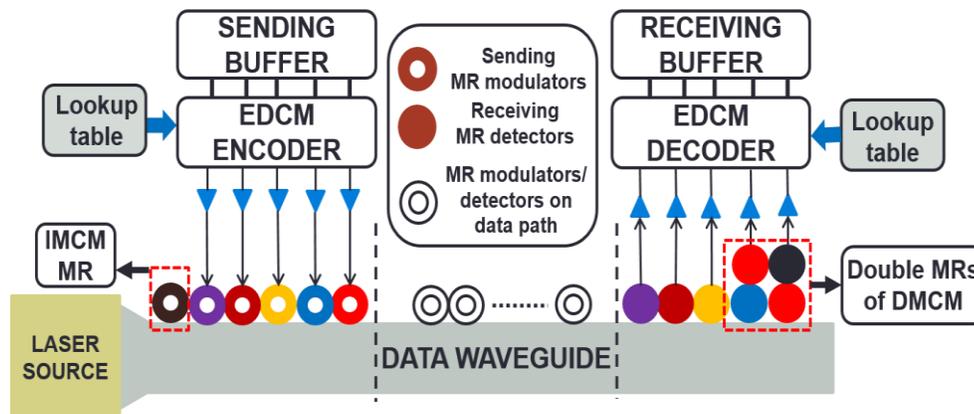


Figure 44 Overview of cross-layer *HYDRA* framework that integrates a device-level IM-aware crosstalk mitigation mechanism (*IMCM*) (see chapter 6), a device-level double MR based crosstalk mitigation mechanism (*DMCM*) and a circuit-level 5-bit crosstalk mitigation mechanism (*EDCM*).

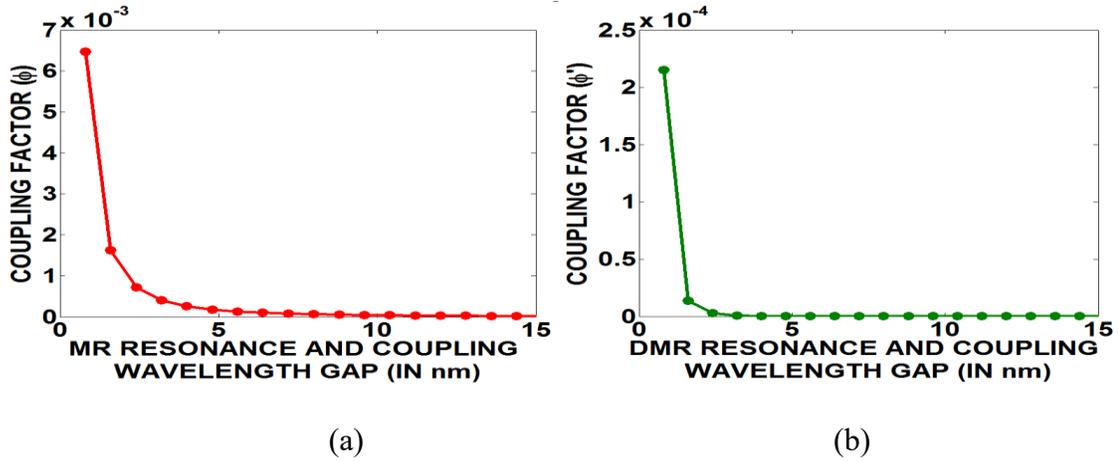


Figure 45 Coupling factor (ϕ/ϕ') variation with increase in gap between the non-resonant wavelength available in the photonic waveguide and the resonance wavelength of (a) a single MR filter and (b) a DMR filter.

7.5. CROSSTALK MITIGATION WITH DMCM

Crosstalk noise in the detectors of DWDM-based PNoCs is mainly caused due to inefficient coupling of filter MRs, as filter MRs in their active mode not only couple photonic power from their resonance wavelengths but also couple a small amount of photonic power from other wavelengths in the waveguide. The coupling factor ϕ in Figure 45(a) represents the fraction of signal power of non-resonant wavelength coupled by an MR filter. This coupled power is then dropped on a detector at the MR's drop port. Figure 45(a) illustrates the variation of ϕ (using Eq. (38)) with increase in gap between the MR resonance wavelength and the non-resonant wavelength available in the waveguide. It can be seen that ϕ decreases abruptly with an increase in this gap. The first immediate non-resonance wavelength has almost $4\times$ higher coupling factor than the second immediate non-resonance wavelength considering a channel spacing of 0.8nm, $Q=12500$, and 5 Gbps bit-rate. We choose these values of channel spacing, Q , and bitrate, as they provide optimal value of total filter penalty for single MR filters (as projected from [123]). *Thus non-*

resonant wavelengths closer to the MR filter's resonance wavelengths create greater crosstalk noise.

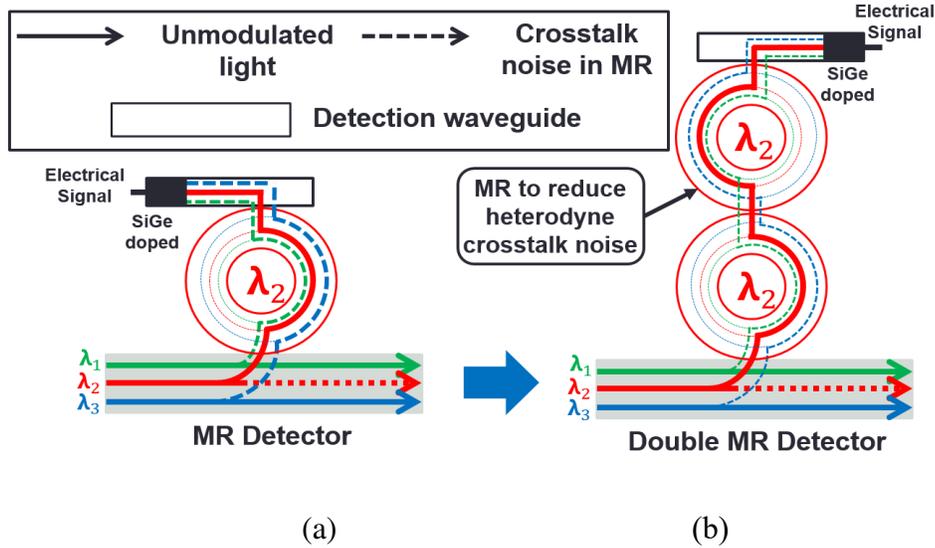


Figure 46 Crosstalk mitigation with double microring resonators: (a) MR detector operation when receiving its resonance wavelength; (b) double MR operation when receiving its resonance wavelength.

One way of reducing this crosstalk noise is to increase the Q-factor of MR filters so that ϕ is reduced. But doing so would increase the photon-lifetime in MR filters limiting their maximum allowable bit-rate [38]. An alternate method for reducing crosstalk is to use second-order filters with double MRs (DMRs), as used in [40] [128], for steeper roll-off of filter response. The use of a DMR filter in place of a single MR filter is depicted in Figure 46. To further reduce crosstalk, use of filter MRs of even higher order (3rd order or higher) is possible, but as explained in [40], the use of higher-order MR filters and the choice of Q for the MR stages trade off crosstalk suppression with signal degradation due to signal side-lobe truncation. From [128], the DMRs present lower signal degradation power penalty than third order and first-order (single MR) MR filters. The optimal crosstalk performance for DMRs is achieved at 12.5Gbps bitrate or lower with

0.8nm channel spacing and the individual MRs having the Q-factor of 8000 [128]. For these reasons, in this chapter we use DMR filters with individual MR Q-factor of 8000 to reduce crosstalk noise.

7.5.1. MODELING OF DMR FILTERS

In this section, we model the resultant coupling factor ϕ' and signal suppression/loss factor γ' due to the steeper roll-off of a DMR filter response. From [129], in analogy to electronic filter design, the effect of steeper roll-off of a DMR filter response can be modeled as a maximally flat Butterworth filter response. From [129], the shape, and hence the Q-factor of the Butterworth filter response does not change for higher order filters (and hence for a DMR) except that the roll-off becomes steeper. Therefore, a Butterworth type of DMR filter response can be modeled by simply setting the exponent of the term $2Q'(\lambda_i-\lambda_j)/\lambda_j$ in Eq. (38) and (39) to four instead of two. As a result, Eq. (38) and (39) can be revised for a DMR to be Eq. (50) and (51), respectively.

$$\Phi'(\lambda_i, \lambda_j, Q') = (1 + (\frac{2Q'(\lambda_i-\lambda_j)}{\lambda_j})^4)^{-1}, \quad (50)$$

$$\gamma'(\lambda_i, \lambda_j, Q') = (1 + (\frac{2Q'(\lambda_i-\lambda_j)}{\lambda_j})^{-4})^{-1}, \quad (51)$$

Here, as the Q-factor for a Butterworth DMR filter does not change from the Q-factor of a single MR filter, Q' in Eq. (50) and (51) can be modeled as the loaded Q-factor of the individual MRs using Eq. (36) and (37). We modeled a DMR with an original Q-factor of 8000 (corresponding to self-coupling coefficients $r_1=0.985$, $r_2=0.985$, and field-transmission coefficient a of 0.985 in Eq. (36) and (37)). Based on this model, we simulated ϕ' using Eq. (50). Figure 45(b) illustrates the variation of ϕ' (using Eq. (50)) with increase in gap between the DMR resonance wavelength and the non-resonant wavelength available in the waveguide. By comparing Figure 45(a) with Figure 45(b), it is evident that ϕ' of the MR's immediate non-resonant wavelength (with

a channel gap of 0.8nm) for the DMR filter is about 30× smaller than ϕ for the single MR filter. Since the coupling factor is used to determine crosstalk noise power in the filter-coupled detectors, it is evident that the DMR filter reduces the crosstalk noise power by about 30×. Thus, it can be concluded that the use of DMR filters in place of signal MR filters at the receiver nodes of PNoCs results in significantly less crosstalk noise power at the detectors. Thus, our double-MR enabled crosstalk mitigation (DMCM) scheme uses DMR filters in place of single MR filters and achieves significant reduction in crosstalk noise power and improvement in OSNR at the detectors.

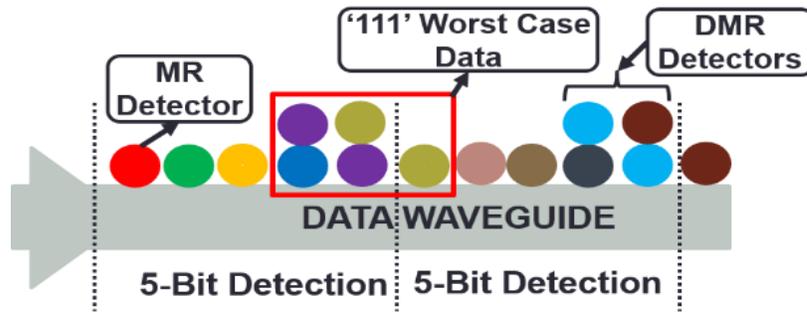


Figure 47 Organization of MR and DMR detectors in a detecting node on a photonic data waveguide with the EDCM mechanism.

7.5.2. OVERHEAD ANALYSIS FOR OUR DMCM SCHEME

In this section, we discuss the overhead of using DMR filters. From [121] and [129], as depicted in Figure 46(b), in a DMR, both the constituent MRs should be in resonance with the same wavelength (i.e., λ_2 in Figure 46(b)) to achieve a smooth filter response without any ripples or multiple peaks. However, in reality, due to the presence of PV, the constituent MRs end up having different resonance wavelengths after fabrication, which results in multiple peaks in the DMR filter's response. Therefore, the resonances of both the individual MRs of a DMR need to be aligned with trimming or tuning, which almost doubles the required trimming or tuning power for DMR filters compared to single MR filters. In addition to this, a DMR filter incurs crosstalk

induced signal impairment related power penalty of 0.5dB for 0.8nm channel spacing [128] and incurs about 1.5dB insertion loss [128]. Moreover, thermal stabilization of a DMR requires 0.9mW more power [128]. Because of all these penalties, too much use of DMR filters result in a very high power overhead. Nevertheless, we propose an intelligent method of using a few DMR filters along with a data encoding mechanism (see next section) to limit the use and overheads of DMRs and further mitigate the crosstalk noise power in the detectors.

Table 18 Code words for EDCM technique

Data Word	Code Word	Data Word	Code Word
0000	00000	1000	01000
0001	00001	1001	01001
0010	00010	1010	01010
0011	00011	1011	01011
0100	00100	1100	10100
0101	00101	1101	10010
0110	10011	1110	10001
0111	10101	1111	10000

7.6. CROSSTALK MITIGATION WITH EDCM

The crosstalk noise in a detector is also highly dependent on the strengths of the non-resonant signals at the detector. Crosstalk noise increases with increase in signal power of non-resonant wavelengths. Based on this observation, one can conjecture that crosstalk noise may be mitigated by placing one or more ‘0’s adjacent to ‘1’s in the data in the waveguide, to reduce photonic signal strength of non-resonant wavelengths. In this section, we present a novel technique (*EDCM*) at the circuit-level for mitigation of crosstalk noise in DWDM-based PNoCs.

DMRs in the *DMCM* technique presented in section 7.5 increase laser power (because of higher signal loss due to higher crosstalk power penalty and insertion loss) and redress power dissipation overheads. These power overheads increase with an increase in the number of DMRs,

hence there is a need to reduce the number of DMRs used with photonic waveguides. In *DMCM*, DMRs are beneficial when there are consecutive ‘1’s in the parallel data word being transmitted, because consecutive ‘1’s imply higher signal strength in the immediate non-resonant wavelengths. One way to reduce the number of DMRs while still minimizing the crosstalk noise due to consecutive ‘1’s is by reducing the number of consecutive ‘1’s in the parallel data word being transmitted. To do so, we propose a circuit-level scheme that employs a sophisticated encoding mechanism.

Our proposed circuit-level DMR-aware crosstalk mitigation mechanism (*EDCM*) places one or more ‘0’s adjacent to ‘1’s in the data to restrict the number of consecutive ‘1’s in the data stream to three. *EDCM* employs 5-bit encoding for every 4-bit data block to restrict the number of consecutive ‘1’s to two in the data block, which in turn limits the worst-case number of consecutive ‘1’s in the data stream to three. Figure 47 shows the organization of MRs and DMRs in the implementation of the proposed *EDCM* encoding mechanism along with the location of occurrence of worst-case consecutive ‘1’s. Table 18 shows the 5-bit codes in the *EDCM* scheme, to replace 4-bit data words. To implement this encoding technique on a 64-bit word, 16 additional bits are required, which in turn increases the number of MR devices by 25%. However, *EDCM* reduces the number of DMR detectors required by *DMCM* and reduces the total number of MR detectors by 12.5%. We propose to use an SRAM based lookup table with a size of 80-bits to facilitate encoding and decoding of data in each modulating and detecting node for our *EDCM* mechanism. This encoding and decoding mechanism incurs a delay overhead of approximately one clock cycle, which we account for in our simulation analysis.

7.7. HYDRA INTEGRATION WITH PNOCS

7.7.1. CORONA PNOCS WITH HYDRA FRAMEWORK

In this subsection, we extend the PV-aware crosstalk models of the Corona PNoC from subsection 7.3.3 to devise PV-aware crosstalk models for Corona enhanced with the *HYDRA* framework. To integrate HYDRA with the Corona PNoC, we increase the DWDM degree in the MWSR waveguides from 64 to 65 (i.e., channel spacing is reduced from 0.8nm to 0.79nm) and increase the number of MWSR waveguides in each channel from 4 to 5 to facilitate simultaneous transfer of an entire packet (which requires 512 bits before encoding). To distribute optical power between these waveguides, there is also a need to replace 1×4 splitters with 1×5 splitters with a splitting factor of R_{S15} . Because of the increase in DWDM from 64 to 65 the number of modulators in the modulating node increases from 64 to 65. Furthermore, we need to add an additional *IMCM* MR in all modulating nodes on each MWSR waveguide, thus the total number of modulators in each modulating node on each MWSR waveguide increases to 66. In the detecting node, first we need to increase the number of detector MRs on each data waveguide from 64 to 65 and secondly as shown in Figure 47 in each group of 5 consecutive detector MRs we need to replace the last two detector MRs with DMR detectors (replace ϕ , and γ with ϕ' , and γ' respectively). Therefore, equations (44), (45), (46), and (47) for worst-case signal and crosstalk noise power are changed to equations (52), (53), (54), and (55) below respectively.

$$P_{signal}(\lambda_j) = \Phi'(\lambda_j, \lambda_j, Q'_{(63 \times 66) + j}) P_S(\lambda_j, \lambda_j), \quad (52)$$

$$P_{noise}(\lambda_j) = \sum_{i=1}^n \Phi'(\lambda_i, \lambda_j, Q'_{(63 \times 66) + j}) (P_S(\lambda_i, \lambda_j)) \quad (i \neq j), \quad (53)$$

$$K(\lambda_i) = (R_{S15})(L_{S15})(L_P)^L (L_B)^B \prod_{n=1}^{63} \prod_{j=1}^{66} \gamma'(\lambda_i, \lambda_j, Q'_{((n-1) \times 66) + j}) \quad (54)$$

$$\psi(\lambda_i, \lambda_j) = \prod_{k=1}^{(k-1) < j} \gamma'(\lambda_i, \lambda_k, Q'_{(63 \times 66) + k}), \quad (55)$$

7.7.2. FIREFLY PNoC WITH HYDRA FRAMEWORK

To investigate the efficacy of integrating our *HYDRA* framework into other PNoC architectures, we integrated it with the Firefly [12] crossbar-based PNoC architecture. Firefly PNoC, for a 256-core system, has 8 clusters (C1-C8) with 32 cores in each cluster. Within each cluster, a group of four cores are connected to a router through a concentrator. Thus each cluster has 8 routers (R1-R8) and these routers are electrically connected using a mesh topology. Firefly uses photonic signals for inter-cluster communication. Unlike the MWSR waveguides used in the Corona crossbar, Firefly uses reservation-assisted single write multiple reader (R-SWMR) data waveguides in its crossbar. Each data channel in Firefly consists of 8 SWMR waveguides, with 64 DWDM in each waveguide. Firefly uses only $1/8^{\text{th}}$ of the MRs on each data waveguide compared to Corona, as only eight nodes are capable of accessing each SWMR waveguide.

In our implementation of Firefly, we considered a power waveguide similar to that used in Corona and determined that the worst-case power loss node (N_{WCPL}) is at the detectors of C4R0, which is the router-0 (R0) of cluster-4 (C4) in this architecture. Similar to Corona, in Firefly, the worst-case signal and noise power in the detectors of router C4R0 are calculated using Eq. (44)-(49) presented in Section 7.3.3. But as Firefly has fewer number of MRs in its data channels, this in turn changes the signal and crosstalk noise power losses before the detector group of N_{WCPL} .

To integrate *HYDRA* with the Firefly PNoC, we need to increase the DWDM degree in SWMR waveguides from 64 to 65 and increase the number of SWMR waveguides in each channel from 8 to 10 to facilitate simultaneous transfer of an entire packet (which requires 512 bits before encoding). To deal with the increase in DWDM degree, we need to increase the number of modulators and detectors from 64 to 65 on each SWMR waveguide in a modulating node and detecting node respectively. Further, we need to add an additional *IMCM* MR in all modulating

and detecting nodes on each SWMR waveguide, which increases the total number of MRs in each modulating and detecting node on each SWMR waveguide to 66. Also, in each detecting node, for each group of 5 consecutive detector MRs (excluding the *IMCM* MR in that detecting node) we need to replace the last two detector MRs with DMR detectors (see Figure 47). Lastly, we determine worst-case OSNR using Eq. (52)-(55) with modified through losses.

7.7.3. FLEXISHARE PNO C WITH HYDRA FRAMEWORK

We also investigated integrating *HYDRA* with the Flexishare [13] PNoC architecture with 256 cores. We considered a 64-radix, 64 node Flexishare architecture with 4 cores in each node having 32 data channels for inter-node communication. Each data channel in Flexishare has four multiple write multiple read (MWMR) waveguides with 64 DWDM in each waveguide. Similar to the MWSR data waveguides of Corona, multiple write multiple read (MWMR) data waveguides in Flexishare also uses the models from Eq. (44)-(49) presented in subsection 7.3.3, to determine the received crosstalk noise and OSNR at detectors for the node with worst-case power loss (N_{WCPL}), which corresponds to detectors of node 63 (R_{63}).

To integrate *HYDRA* with Flexishare, we need to increase the DWDM degree in the MWMR waveguides from 64 to 65 and increase the number of MWMR waveguides in each channel from 4 to 5 to simultaneously transfer 512 bits. We also need to increase the number of modulators and detectors from 64 to 65 on each MWMR waveguide in each modulating and detecting node. Similar to the Firefly PNoC, we need to add an additional *IMCM* MR in all modulating and detecting nodes on each MWMR waveguide, which increases the total number of MRs in each modulating and detecting node on each SWMR waveguide to 66 respectively. In the detecting nodes of Flexishare, for each group of 5 consecutive detector MRs (excluding the *IMCM* MR in

that detecting node), we need to replace the last two detector MRs with DMR detectors. Lastly, we can use Eq. (42)-(55) to determine worst-case OSNR.

7.8. EVALUATION

7.8.1. SIMULATION SETUP

To evaluate the efficacy of our proposed cross-layer crosstalk noise mitigation framework *HYDRA* which combines device layer (*IMCM*, *DMCM*) and circuit layer (*EDCM*) mechanisms for DWDM-based PNoCs, we integrate the framework with the Corona, Firefly, and Flexishare crossbar-based PNoCs, as explained in Section 7.7. We modeled and performed simulation based analysis of the *HYDRA*-enhanced Corona, Firefly, and Flexishare PNoCs using a cycle-accurate SystemC based NoC simulator, for a 256-core single-chip architecture at 22nm. We validated the simulator in terms of power dissipation and energy consumption based on the results obtained from the DSENT tool [75]. We used real-world traffic from applications in the PARSEC benchmark suite [43]. GEM5 full-system simulation [72] of parallelized PARSEC applications was used to generate traces that were fed into our cycle-accurate NoC simulator. We set a “warmup” period of 100 million instructions and then captured traces for the subsequent 1 billion instructions. These traces are extracted from parallel regions of execution of PARSEC benchmark applications. We performed geometric calculations for a 20mm×20mm chip size, to determine lengths of MWSR, SWMR, and MWMR waveguides in the Corona, Firefly, and Flexishare PNoCs. Based on this analysis, we estimated the time needed for light to travel from the first to the last node as 8 cycles at 5 GHz clock frequency [25] [67]. We use a 512-bit packet size, as advocated in the Corona, Firefly, and Flexishare PNoCs.

The static and dynamic energy consumption of electrical routers and concentrators in Corona, Firefly, and Flexishare PNoCs is based on results from the open source DSENT tool [75]. We model and consider area, power, and performance overheads for our framework implemented with the Corona, Firefly, and Flexishare PNoCs, as follows. *HYDRA* with Corona, Firefly, and Flexishare PNoCs has an electrical area overhead estimated to be 6.4 mm², 12.7 mm², and 3.4 mm² respectively and power overhead of 0.23 W, 0.44 W, and 0.36 W respectively, using gate-level analysis and the CACTI 6.5 [114] tool for memory and buffers. The photonic area overhead of Corona, Firefly, and Flexishare architecture is 9.63 mm², 19.83 mm², and 5.2 mm² respectively, based on the physical dimensions [104] of their waveguides, MRs, and splitters. For energy consumption of photonic devices, we adapt model parameters from recent work [73], [74], [115], with 0.42pJ/bit for every modulation and detection event and 0.18pJ/bit for the driver circuits of modulators and photodetectors. The MR trimming power is set to 130μW/nm [18] for current injection (blue shift) and tuning power is set to 240μW/nm [118] for heating (red shift).

7.8.2. WORST-CASE OSNR COMPARISON FOR VARIOUS PNOCS

Our first set of simulation results compares the baseline (without any crosstalk-mitigating enhancements) Corona, Firefly and Flexishare PNoCs with four variants of these architectures corresponding to three crosstalk-mitigation strategies from prior work (PCTM5B and PCTM6B from [28], PICO from [31]) and our proposed *HYDRA* framework from this chapter. PCTM5B and PCTM6B are encoding schemes that replace each 4-bits of a data word with 5-bit and 6-bit code words respectively. These schemes aim to reduce photonic signal-strength of immediate non-resonant wavelengths (adjacent wavelengths in DWDM) to decrease crosstalk and improve OSNR in MR detectors. PICO is a process-variation aware crosstalk mitigation mechanism which also

encodes data to reduce photonic signal-strength of immediate non-resonant wavelengths based on the process variation profile of the receiving MR detectors.

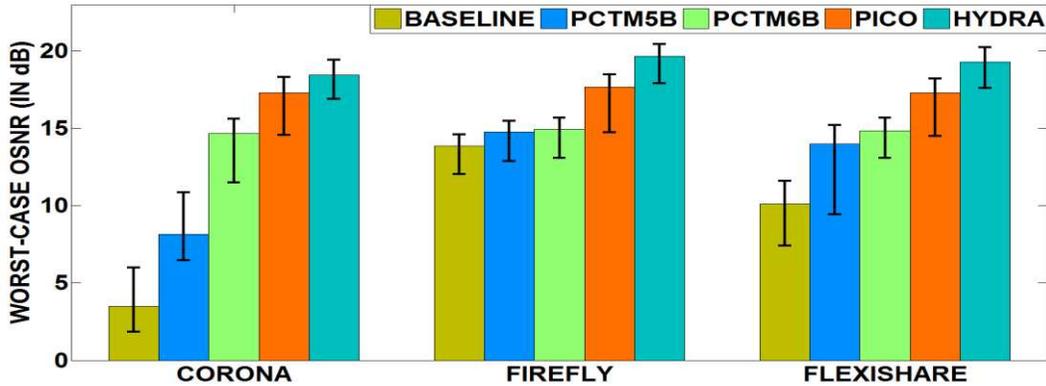


Figure 48 Worst-case OSNR comparison of *HYDRA* with PCTM5B [28], PCTM6B [28], and PICO [31] for Corona, Firefly, and Flexishare PNoCs. Bars show mean values of worst-case OSNR across 100 PV maps; confidence intervals show variation in worst-case OSNR.

Utilizing the models presented in Sections 7.3 and 7.7, we calculate the received crosstalk noise and OSNR at detectors for the node with worst-case power loss (N_{WCPL}), which correspond to MR detectors in cluster 64 for the Corona PNoC, MR detectors of router C4R0 for the Firefly PNoC, and MR detectors of node R₆₃ for the Flexishare PNoC. Figure 48 summarizes the worst-case OSNR results for the baseline, PCTM5B, PCTM6B, PICO, and *HYDRA* configurations of the three PNoC architectures considered. From the figure, it can be observed that Corona PNoC with *HYDRA* has 5.3×, 2.26×, 1.25×, and 1.06×, Firefly PNoC with *HYDRA* has 1.42×, 1.33×, 1.32×, and 1.13×, and Flexishare PNoC with *HYDRA* has 1.96×, 1.41×, 1.33×, and 1.14× worst-case OSNR improvements on average, compared to the baseline and PCTM5B, PCTM6B, and PICO enhanced variants of these architectures respectively.

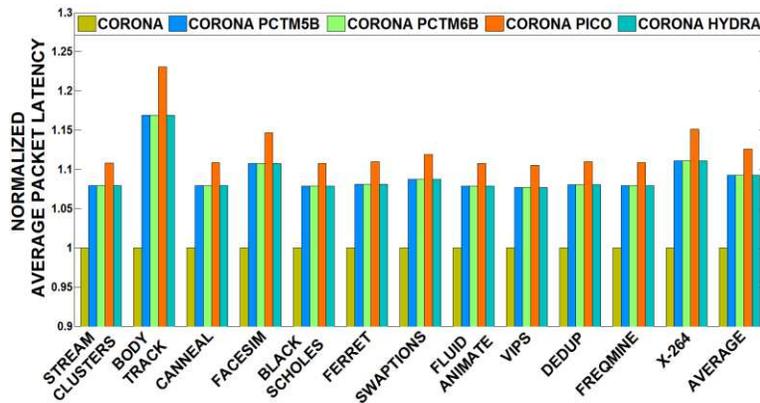
Both PCTM5B and PCTM6B eliminate occurrences of ‘111’ in a data word and have limited occurrences of ‘11’, which helps to reduce crosstalk noise in the detectors. But these techniques

are unable to eliminate all occurrences of ‘11’, because of which these techniques are unable to achieve higher reduction in crosstalk noise and significant improvement in OSNR. PICO considers the PV-profile of detecting nodes and performs encoding on specific wavelengths where there is high signal loss due to trimming to reduce crosstalk noise and improve OSNR in PNoCs. But even with the PICO technique, there still exist occurrences of ‘111’ and ‘11’, because of which OSNR gains with PICO are on the lower side. In contrast, *HYDRA* virtually eliminates all of the occurrences of ‘111’ and ‘11’ from the data word by combining benefits from *IMCM* and *DMCM*, and using *EDCM*’s 5-bit encoding mechanism. Although *EDCM*’s 5-bit encoding still results in limited occurrences of ‘111’ and ‘11’ in a data word, the DMRs of *DMCM* reduce the impact of consecutive ‘1’s in the data word by removing crosstalk noise generated by these ‘1’s in detector MRs. Thus *HYDRA* demonstrates higher OSNR gains compared to the best known previously proposed techniques. Furthermore, the OSNR values achieved with *HYDRA* (see Figure 48) are sufficient to enable reliable data transfers in PNoCs such as Corona, Firefly, and Flexishare.

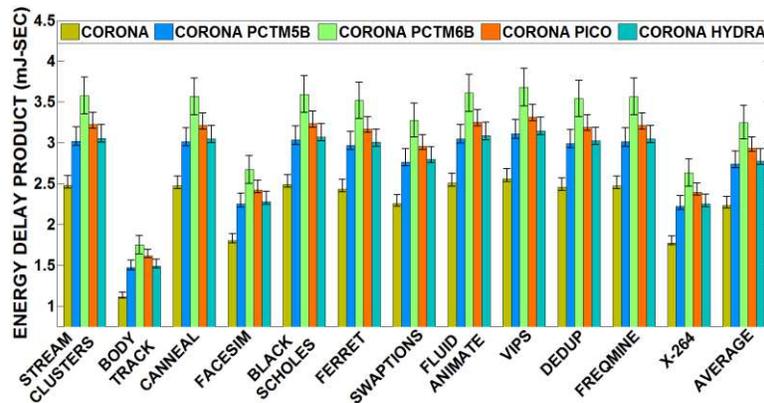
7.8.3. OVERHEAD ANALYSIS OF HYDRA WITH VARIOUS PNOCS

Our last set of results quantify the overhead for the proposed *HYDRA* framework and other techniques when used with the Corona, Firefly, and Flexishare PNoCs. Figure 49(a) and Figure 49(b) present detailed simulation results that quantify the average network packet latency and energy-delay product (EDP) for five Corona configurations. Results are shown for 12 multi-threaded PARSEC benchmarks. From Figure 49(a) it can be seen that on average, Corona with *HYDRA* has 9.24% higher latency compared to the baseline. The additional delay due to encoding and decoding of data with *HYDRA*, PCTM5B, PCTM6B, and PICO contributes to their increase in average latency. The penalty due to encoding/decoding is approximately 1 cycle in PCTM5B,

PCTM6B, and *HYDRA*. Thus *HYDRA* has a similar overhead compared to PCTM5B and PCTM6B. However, *PICO* has a 2 cycle penalty, which increases its delay compared to *HYDRA* by 3.1%. Note that for the chosen clock frequency, PV in photonic components does not change the number of clock cycles for various operations, such as encoding/decoding, modulation/detection etc., therefore Figure 49(a) does not have confidence intervals or variations in packet latency due to PV.



(a)



(b)

Figure 49 (a) Normalized average latency and (b) energy-delay product (EDP) comparison between Corona baseline and Corona configurations with PCTM5B, PCTM6B, PICO, and *HYDRA* techniques, for PARSEC benchmarks. Latency results are normalized to the baseline Corona results. In the EDP plot, bars represent mean values of EDP across 100 PV maps; confidence intervals show variation in EDP.

From the results for EDP shown in Figure 49(b), it can be seen that on average, the Corona configuration with our *HYDRA* framework has 24.3% higher EDP compared to the baseline. The increase in EDP for Corona with *HYDRA* is not only due to the increase in average latency, but also due to the addition of extra bits for encoding and decoding, which leads to an increase in the amount of photonic hardware in the architectures (more number of MRs, complex splitters). This in turn increases static power dissipation. Dynamic power also increases in these architectures, but by much less amount. However, EDP for Corona with *HYDRA* is 17.1% and 5.7% lower compared to PCTM6B and PICO respectively. The higher latency of PICO compared to *HYDRA* increases its EDP, whereas *HYDRA* has lower EDP compared to PCTM6B because *HYDRA* conserves laser and MR trimming/tuning power due to a lower photonic hardware footprint compared to PCTM6B. The EDP for Corona with *HYDRA* is 1.3% higher compared to PCTM5B. Although PCTM5B and *HYDRA* have similar average latency, the increase in number of MRs in *HYDRA* due to the presence of *IMCMMRs* and *DMRs* increases its laser and trimming/tuning power, which in turn increases its EDP.

Figure 50(a) and Figure 50(b) summarize the average network packet latency and EDP results for the five configurations of Firefly and Flexishare PNoCs. Results are shown for twelve multi-threaded PARSEC benchmarks and are averaged across these benchmark applications, for brevity. From Figure 50(a) it can be observed that on average, Firefly with *HYDRA* has 5.2% and Flexishare with *HYDRA* has 10.6% higher latency compared to their respective baselines. The additional delay due to encoding and decoding of data with *HYDRA* contributes to its increase in average latency over the respective baselines of Firefly and Flexishare PNoCs. The latency overhead for Firefly with *HYDRA* is lower compared to Corona and Flexishare with *HYDRA*. This is because Firefly is a hybrid PNoC where some portion of data traverses through electrical links.

This data over electrical links is unaffected by the extra encoding/decoding delays in *HYDRA*, whereas in Corona and Flexishare the entire traffic traverses through photonic waveguides. Much like Corona (Figure 49(a)), the Firefly and Flexishare architectures with *HYDRA* have similar latency values compared to these architectures with PCTM5B and PCTM6B (Figure 50(a)). Furthermore, Firefly with *HYDRA* has 2.7% and Flexishare with *HYDRA* has 3.2% lower latency compared to PICO. Reduction in number of encoding or decoding cycles from 2 to 1 from PICO to *HYDRA* reduces average latency of *HYDRA*.

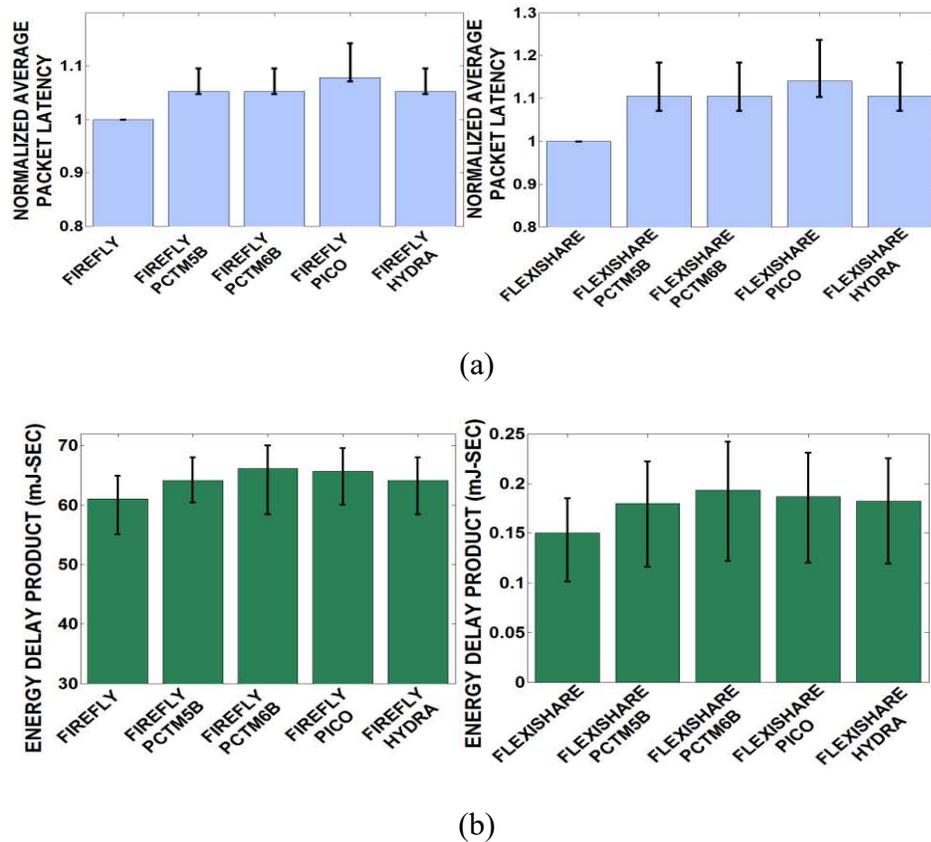


Figure 50 (a) Normalized average latency and (b) energy-delay product (EDP) comparison between different variants of Firefly and Flexishare PNoCs which include their baselines and their variants with PCTM5B, PCTM6B, PICO, and *HYDRA* techniques, for PARSEC benchmark applications. Latency results are normalized with their respective baseline architecture results. Bars represent mean values of average latency and EDP for 100 PV maps; confidence intervals show variation in average latency and EDP across PARSEC benchmarks.

From the results for EDP shown in Figure 50(b), it can be seen that on average, the Firefly and Flexishare configurations with our *HYDRA* framework have 5% and 22% higher EDP compared to their respective baselines. EDP overhead for Firefly is relatively lower compared to the Corona and Flexishare architectures because of its lower latency overheads and smaller increase in laser/trimming power due to lesser increase in the amount of photonic hardware. Firefly with *HYDRA* has 3.1% and 2.4%, and Flexishare with *HYDRA* has 5.9% and 2.4% lower EDP compared to the respective architecture configurations with PCTM6B and PICO. Additionally, compared to Firefly and Flexishare configurations with PCTM5B, the configurations of the same architectures with *HYDRA* framework have 0.6% and 1.5% higher EDP respectively.

7.9. CONCLUSIONS

We have presented a novel cross-layer crosstalk mitigation framework for the reduction of crosstalk noise in the detectors of DWDM-based PNoC architectures. Our proposed *HYDRA* framework seamlessly integrates two device layer and a circuit layer technique to enable interesting trade-offs between reliability, performance, and energy overheads for the Corona, Firefly, and Flexishare crossbar-based PNoC architectures. Our simulation based analysis shows that the *HYDRA* framework improves worst-case OSNR by up to $5.3\times$ compared to the baseline architectures, and by up to $1.14\times$ compared to the best known PNoC crosstalk mitigation scheme from prior work. Thus, *HYDRA* is an attractive solution to enhance reliability in emerging DWDM-based PNoCs.

8. ISLANDS OF HEATERS: A NOVEL THERMAL MANAGEMENT FRAMEWORK FOR PHOTONIC NOCS

Operation of photonic NoCs (PNoCs) is very sensitive to temperature variations that frequently occur on a chip. These variations can create significant reliability issues for PNoCs. For example, microring resonators (MRs) which are the building blocks of PNoCs, may resonate at another wavelength instead of their designated wavelength due to thermal variations, which can lead to bandwidth wastage and data corruption in PNoCs. This chapter proposes a novel run-time framework to overcome temperature-induced issues in PNoCs. The framework consists of (i) a PID controlled heater mechanism to nullify the thermal gradient across PNoCs, (ii) a device-level thermal island framework to distribute MRs across regions of temperatures; and (iii) a system-level proactive thread migration technique to avoid on-chip thermal threshold violations and to reduce MR tuning/trimming power by migrating threads between cores. Our experimental results with 64-core Corona and Flexishare PNoCs indicate that the proposed approach reliably satisfies on-chip thermal thresholds and maintains high network bandwidth while reducing total power by up to 64.1%.

8.1. MOTIVATION AND CONTRIBUTION

Photonic components and especially MRs are extremely susceptible to thermal fluctuations. Figure 51 depicts the impact of thermal variation on MRs. MRs R_1 - R_n have been designed to resonate on wavelengths λ_1 - λ_n respectively at temperature T_1 . As the temperature increases, due to the resulting variations in refractive index, each MR now resonates with a different wavelength towards the red end of the visible spectrum (i.e., red-shift). This red-shift is shown in the figure

where, at temperature T_2 , MR R_i will now be in resonance with λ_{i-1} . This phenomenon reduces transmission reliability and results in wastage of available bandwidth, e.g., MRs are unable to read or write to wavelength λ_n at temperature T_2 .

Maintaining a uniform temperature across all the MRs is a must for reliable data transmission in PNoCs. But thermal fluctuations and gradients are common in CMPs. 3D-ICE [130] simulations of PARSEC [43] and SPLASH-2 [131] benchmarks indicate a 15-20K peak thermal gradient in a 64-core CMP as shown in Figure 52. Such a huge gradient causes a mismatch of resonant wavelengths of MRs, leading to unreliable data transmission and PNoC performance degradation.

Recently, few techniques have been proposed to address thermal issues in PNoCs. At the *device-level*, a trimming mechanism is proposed in [18] that induces a blue shift (decrease) in the resonance wavelengths of MRs using carrier injection. A tuning technique was demonstrated in [19] where a red-shift (increase) in the resonance wavelengths is induced by using a localized heater. Further several athermal photonic devices have been presented to reduce the localized tuning/trimming power in MRs. These design time solutions include using cladding to reduce thermal sensitivity [132] and using heaters as well as temperature sensors for thermal control. While these device-level techniques are promising, they either possess a high power overhead or require costly changes in the manufacturing process (e.g., much larger device areas) that would decrease network bandwidth density and area efficiency. At the *system-level*, a thread migration framework was presented in [33] to avoid on-chip thermal threshold violations and also reduce trimming/tuning power for MRs. In [133], a ring aware thread scheduling policy was proposed to reduce on-chip thermal gradients in a PNoC. A proportional-integral-derivative (PID) heater mechanism was proposed in [134] that minimizes the effect of thermal variation on PNoC's

performance and power. However, all these system-level techniques do not consider the impact of run-time workload variations and also result in considerable power performance overheads.

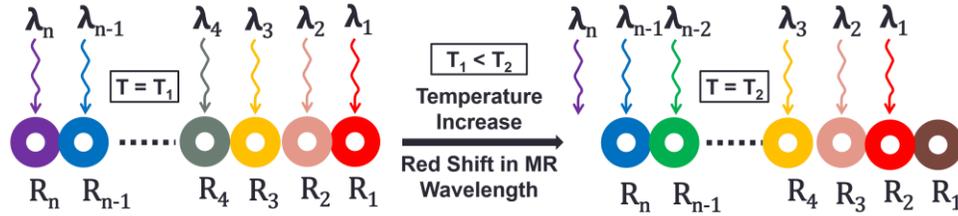


Figure 51 Impact of thermal variations on MRs.

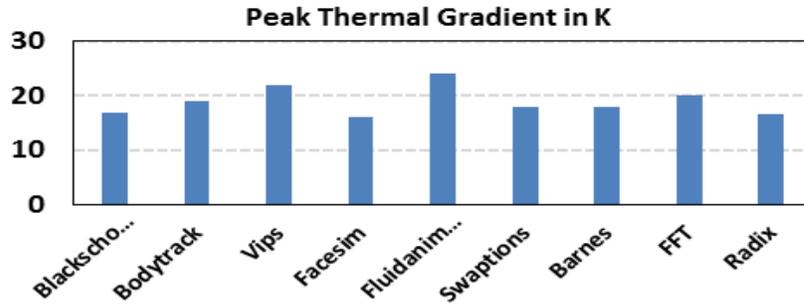


Figure 52 Peak thermal gradient (in Kelvin) across a 64-core chip running 48-threaded PARSEC [43] and SPLASH-2 [131] benchmarks.

Our goal in this chapter is to minimize thermal variations with reduced localized thermal tuning and trimming in PNoCs, thereby reducing key overheads and ultimately easing the adoption of PNoCs for future CMP systems. We propose a novel low-power thermal management framework that integrates an adaptive heater mechanism at the device-level and a dynamic thread migration scheme at the system-level. This chapter makes the following contributions:

- A novel temperature island framework with adaptive heater based MR to handle thermal gradients across PNoC;

- An islands of heaters based dynamic thread migration (IHDTM) scheme in conjunction with a support vector regression based temperature prediction mechanism. Such a scheme nullifies on-chip thermal threshold violations and also reduces trimming/tuning power for MRs;
- The evaluation of the proposed framework on a 64-core CMP with a system-level simulator shows: (a) 70% improvement in trimming power dissipation over the most recent prior work, (b) 64.1% improvement in total power dissipation compared to a state-of-the-art thermal management technique, (c) 13.72K improvement in peak temperature, and (d) these improvements are achieved while maintaining full network-bandwidth.

The rest of the chapter is organized as follows. Section 8.2 explains the proposed thermal management framework in detail. Experiments, results, and comparative analysis are demonstrated in Section 8.3 followed by conclusions in Section 8.4.

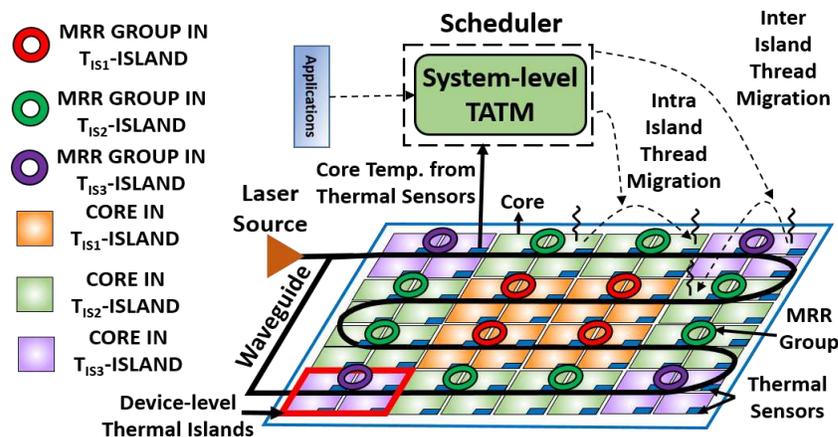


Figure 53 IHDTM framework with device-level thermal islands and system-level temperature-aware thread migration mechanism (TATM).

8.2. ISLANDS OF HEATERS BASED DYNAMIC THERMAL MANAGEMENT (IHDTM)

The proposed IHDTM framework enables variation-aware thermal management by integrating device-level and system-level enhancements. A high-level overview of the framework

is shown in Figure 53. At the device-level, the entire PNoC layer is divided into ‘ k ’ regions or islands, namely: T_{IS1} -island, T_{IS2} -island, T_{IS3} -island, and so on. All MRs in the T_{ISi} -island ($i \leq k$) are designed to operate at T_{ISi} ; similarly, MRs in the other islands are designed to operate at their respective temperatures. We use our device-level technique to overcome small deviations ($\pm 10K$) in T_{ISi} whereas the system-level technique is used to adapt to larger variations ($> \pm 10K$). The device-level technique aims to adapt to the changing on-chip thermal profile, maintaining maximum bandwidth and correct MR operation while minimizing trimming and tuning power in the PNoC. At the system-level, the dynamic thread migration scheme maintains acceptable core-temperatures for each island. The following sections explain the proposed (i) device-level island framework and (ii) system-level thread migration scheme in detail.

8.2.1. THERMAL ISLANDS

The thermal distribution across a 64-core PNoC chip running PARSEC and SPLASH-2 benchmarks (using 3D-ICE simulation) shows three major zones of temperature: 363K, 343K, and 323K. Also, the average thermal gradient in the PNoC chip is found out to be approximately 15-20K. To reduce this gradient, the proposed device-level framework adopts three islands (as shown in Figure 53) each of which are maintained at a unique temperature by assigning T_{IS1} , T_{IS2} , and T_{IS3} to 363K, 343K, and 323K respectively. As mentioned in the previous section, MRs in the 363K-island (T_{IS1} -island) are designed to operate at 363K with a variation range of $\pm 10K$. MRs employ thermal tuning and electrical trimming when they are operated below and above their designed temperatures respectively. Similarly, MRs in other islands are designed to operate at the respective temperatures. For PNoCs of other sizes (e.g. 16-core, 25-core, 36-core, 128-core, 256-

core), there can be slight variations in the number of islands and their respective temperature zones. Accordingly, the numbers of islands and their temperatures can be fixed at design time.

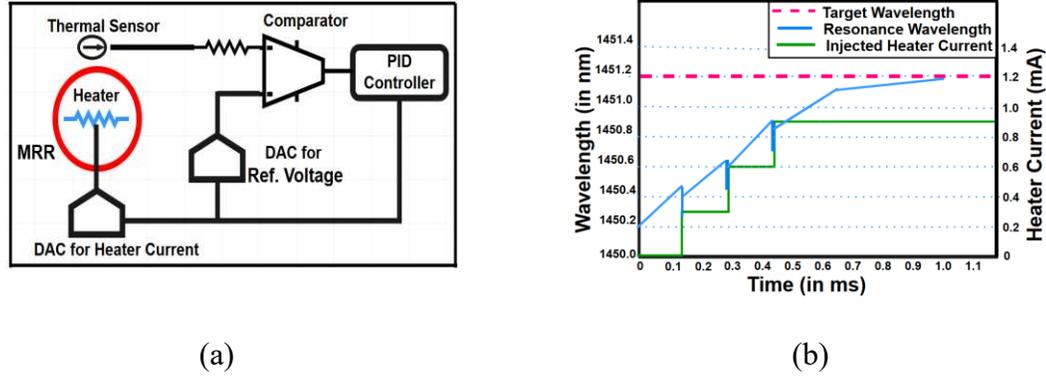


Figure 54 (a) MR with adaptive heater (b) Thermal tuning of MR

Algorithm 2 Thermal management of MR

Input: Temperature (T) around the MR detected by thermal sensor

//Controller converts T to appropriate heater current as follows:

- 1: $dT = |T_{island} - T|$
- 2: $P_{Heat} = \frac{dT}{\rho} \times H_{eff}$
- 3: **if** ($T \leq T_{island}$) **then** $i_{Heat} = i_{Max} - \sqrt{\frac{P_{Heat}}{R_{Heat}}}$
- 4: **else** $i_{Heat} = i_{Max} + \sqrt{\frac{P_{Heat}}{R_{Heat}}}$

Output: current (i_{Heat}) to be fed to heater

To manage localized temperature variation below designed temperature, each MR is integrated with a PID controller [15] based heater as shown in Figure 54(a). The PID controller is tuned with proportional band $K_p=50$, integral cycle-time $K_i=1$ millisecond (ms), and derivative coefficient $K_d=0$. An open source PID tuning software [135] is used to determine optimal values of K_p , K_i , and K_d .

Algorithm 2 depicts the control algorithm for the heater in each MR to stabilize thermal variations. In the algorithm, T represents the temperature across an MR as detected by the

corresponding thermal sensor, T_{island} is the fixed temperature of the island in which the MR resides ($T_{island} = T_{ISI}$), P_{Heat} is the heater power, i_{Heat} represents heater current, and H_{eff} stands for the transfer function of the heater. With any local temperature change dT , there is an equivalent shift in resonance for the MR. To undo this resonance shift in an MR, an equivalent amount of heat must be radiated by the heater integrated with that MR. As per the algorithm, the controller collects temperature data T from the local thermal sensor as input. In step 1, the absolute value of the difference between T and T_{island} is calculated followed by determining the required heater power P_{Heat} in step 2. T is compared with T_{island} in step 3 and accordingly the required heater current i_{Heat} is computed either in steps 3-4. The evaluated value of i_{Heat} is fed to the heater coil. This amount of current is needed by the heater to maintain the fixed temperature T_{island} around the MR. Our analysis shows that a maximum of 1 ms of time is needed for the heater element to bring the surrounding temperature to the desired value of T_{island} . We account for this time delay in our simulations. Figure 54(b) shows the tuning process of an MR with injected heater current as explained in the algorithm. The control algorithm is invoked after every 1ms for each MR.

This heater-based technique helps to stabilize thermal fluctuations in each temperature island with reduced tuning power. However, if the power footprint of a workload on a core associated with a 363K-island is very low, its core temperature may fall below the lower thermal limit (i.e. smaller than 353K). This thermal gradient can significantly increase tuning power consumption of an associated MR. Similarly, if the power footprint of a workload on a core associated with a 323K-island keeps increasing beyond a threshold, then its core temperature might reach beyond the control of the MR-trimmer (i.e. greater than 333K). This will in turn permanently shift the resonance of the MR, inducing errors during communication. To address these issues, we propose a system-level temperature-aware thread migration (TATM) technique that performs thread

migration to idle cores to maintain temperatures of corresponding MRs close to the design-temperatures of their respective islands. By intelligently migrating threads, this technique reduces device-level tuning/trimming power in MRs. TATM also aims to proactively reduce thermal hotspots, which in turn will reduce instances of irrecoverable drift in MRs.

Table 19 List of TATM parameters and their definitions

Symbol	Definition
IPC_i	Instructions per cycle of i^{th} core
T_i	Current temperature of i^{th} core
TN_i	Average temperature of immediate neighboring cores of i^{th} core; if this core is on chip periphery and missing neighbors, then we consider virtual neighbor cores at ambient temperature in lieu of the missing cores
PT_i	Predicted temperature of i^{th} core
T_{ij}	Thermal threshold of T_{ISj} -island
T_{lj}	Thermal limit of T_{ISj} -island
$IEIC_{ij}$	Inter-island cores for T_{ISj} -island whose island MRs design temperature is greater than T_{ISj}
$IEIC_{ij}$	Inter-island cores for T_{ISj} -island whose island MRs design temperature is smaller than T_{ISj}
$IAIC_j$	Intra-island cores for T_{ISj} -island
C	Regularization parameter
W	Weight vector for regression
x_i and y_i	Input and outputs in training and test data
ξ_i	Slack variables
E	Error function
b	Bias for cost function

8.2.2. TEMPERATURE-AWARE THREAD MIGRATION SCHEME (TATM)

8.2.2.1. OBJECTIVE

The primary goal with TATM is to maintain the temperature of all the cores in an island on a die below a specified thermal threshold (T_i) and above a thermal limit (T_l), i.e., for a core i in the T_{ISj} -island, $T_{lj} \leq T_i \leq T_{ij}$ where T_i is the temperature of core i , T_{ij} is threshold temperature of T_{ISj} -island, and T_{lj} is thermal limit of T_{ISj} -island. TATM maintains the core temperatures such that the

temperature of all the MRs within an island is close to their design temperature, to reduce tuning power consumption in adaptive heaters as explained in the previous section.

We utilize support vector based regression (SVR) to predict the future temperature of a core. This predicted temperature of a core is compared with the corresponding island’s thermal threshold (upper limit) and thermal limit (lower limit) to determine the potential for a thermal emergency. If such a potential exists, then TATM initiates thread migration. Inter-island thread migration (Inter-island cores (*IEIC*)) is preferred over intra-island thread migration (Intra-island cores (*IAIC*)). This step has a twofold benefit. Firstly, by moving the thread away from a core that could suffer a thermal emergency, we avoid instances of irrecoverable drift in the MR groups of that core. Secondly, by moving the thread to a core in different island, we ensure that the temperature of the island and its corresponding ring blocks remains between the island’s thermal threshold (T_{i1} , T_{i2} , and T_{i3}) and thermal limit (T_{l1} , T_{l2} , and T_{l3}) to conserve trimming/tuning power. If a thermal emergency occurs due to exceeding the thermal threshold, then it is preferred that the thread is migrated to a core in an island whose MR design temperature is higher. If a thermal emergency occurs due to temperature falling below the thermal limit then it is preferred that the thread is migrated to a core in an island whose MR design temperature is lower. The parameters used to describe TATM in this section are shown in Table 19.

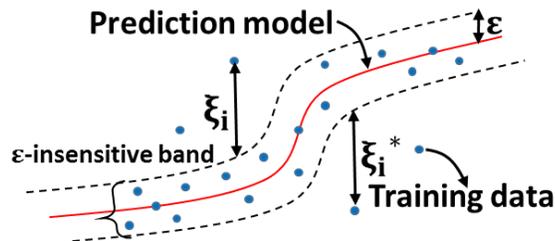


Figure 55 Non-linear support vector based regression prediction model.

8.2.2.2. TEMPERATURE PREDICTION MODEL

We designed a support vector regression (SVR) based temperature predictor that accepts input parameters reflecting the workload for a core i , in terms of instructions per cycle (IPC_i), temperature (T_i), and surrounding core temperatures (T_{Ni}), and predicts the future temperature for core i .

Architecture: A typical SVR [136], [137] relies on defining a prediction model that ignores errors that are situated within the ε range of the true value. This type of a prediction model is called an ε -insensitive prediction model. Figure 55 shows an example of a one-dimensional non-linear SVR based prediction model with an ε -insensitive band. The variables (ξ and ξ^*) measure the cost of the errors on the training points. These are zero for all points that are inside the ε -insensitive band.

SVR is primarily designed to perform linear regression. To handle non-linearity in data, SVR first maps the input x_i onto an m -dimensional space using some fixed (non-linear) mapping notated as Φ , and then a linear model is constructed in this high-dimensional space as shown in Eq. (57) and (58) below. Thus, it overcomes drawbacks of linear and logistic regression towards handling non-linearity in data. This class of SVRs is called *kernel based SVRs* which use kernel κ as shown in Eq. (59) for implicit mapping of non-linear training data (as shown in Figure 55) into a higher dimensional space.

$$CF = \min \frac{1}{2} W^T \cdot W + C \sum_{i=1}^n (\xi_i + \xi_i^*) \quad (56)$$

Subject to:

$$y_i - W^T \Phi(x_i) - b \leq \varepsilon + \xi_i \quad (\xi_i \geq 0, i = 1, 2, \dots, n) \quad (57)$$

$$W^T \Phi(x_i) + b - y_i \leq \varepsilon + \xi_i^* \quad (\xi_i^* \geq 0, i = 1, 2, \dots, n) \quad (58)$$

$$\kappa(x_i, x_j) = \Phi(x_i)^T \Phi(x_j) \quad (59)$$

SVR performs linear regression in this high-dimension space using ε -insensitive loss and, at the same time, tries to reduce model complexity by minimizing $W^T \cdot W$. This can be described by introducing (non-negative) slack variables ξ_i and ξ_i^* ($i = 1$ to n), to measure the deviation of training samples outside the ε -insensitive band. Thus SVR is formulated as minimization of the cost function (CF) in Eq. (56) with constraints shown in Eq. (57) and (58).

As on-chip temperature variation data is non-linear in the original space, our SVR model employs a kernel based regression which uses a Radial Basis Function (RBF) [138] (Gaussian kernel) as shown in Eq. (60). The RBF kernel improves the accuracy of SVR when data has non-linearity in the original space. We performed a sensitivity analysis (SA) to determine regularization parameter (C) and ‘gamma’ (γ) values of the kernel based SVR (see Section 8.3.1 for chosen values). This SA overcomes the possibility of over fitting of training data and improves accuracy further.

$$\kappa(x_i, x_j) = \exp\left(-\gamma |x_i - x_j|^2\right) \quad (60)$$

Training and Accuracy: We trained our SVR model using a set of multi-threaded applications from the PARSEC [43] and SPLASH-2 [131] benchmark suites, specifically: blackscholes (BS), bodytrack (BT), vips (VI), facesim (FS), fluidanimate (FA), swaptions (SW), barnes (BA), fft (FFT), radix (RX), radiosity (RD), and raytrace (RT) with different thread counts: 2, 4 and 8. We considered different combinations of thread mappings on a 9-core (3×3) floorplan, to train our predictor to determine the temperature of the center (target) core. The threads mapped to a 9-core floorplan represents a generic mapping and can be applied to 64-core, 128-core, and 256-core floorplans.

As the future temperature of a target core is dependent on the average temperature of its immediate neighboring cores, we trained our SVR model with temperature inputs from the target

core running a single thread, as well as its surrounding cores running a variable number of threads. Simulations with various mappings of these threads allowed us to obtain data to train our SVR model. This data included temperature for the target core and its neighboring core temperatures, as well as instructions per cycle (*IPC*) for the target core. *IPC* is very useful to determine if there is a phase change in an application and plays a crucial role in maintaining future temperature prediction accuracy especially when temperatures of a target core and its neighbors are similar at a given time. Our training algorithm involved an iterative process that adjusts the weights and bias values in the SVR (Eq. (56)-(58)) to fit the training set.

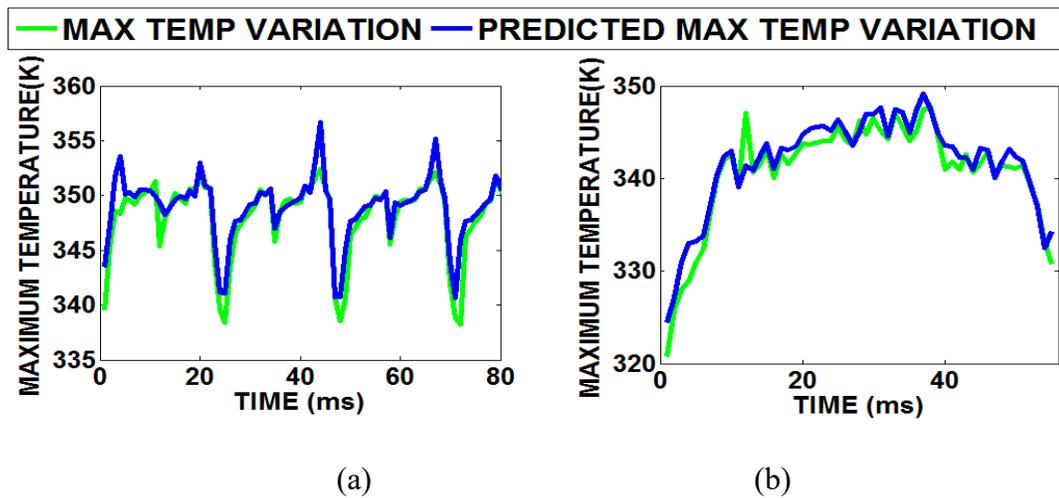


Figure 56 Actual and predicted maximum temperature variation with execution time for (a) fluidanimate (FA) and (b) radiosity (RD) benchmarks run on a 64-core platform executing 32-threads.

We verified the accuracy of our SVR model for multi-threaded benchmark workloads (we considered 6000 floorplans, with 70% of input data for training and 30% for testing) and found that it has an accuracy of over 95%. Figure 56(a) and (b) show actual and predicted on-chip temperature variations for a 64-core platform executing 32 threads of the FA and RD benchmarks. From these figures it can be seen that our temperature predictor tracks temperature quite accurately.

When predicted temperature is beneath thermal limit or exceeds the thermal threshold our thread migration mechanism (which is discussed next) migrates threads between cores to reduce tuning/trimming power and keep overall peak temperature below the threshold.

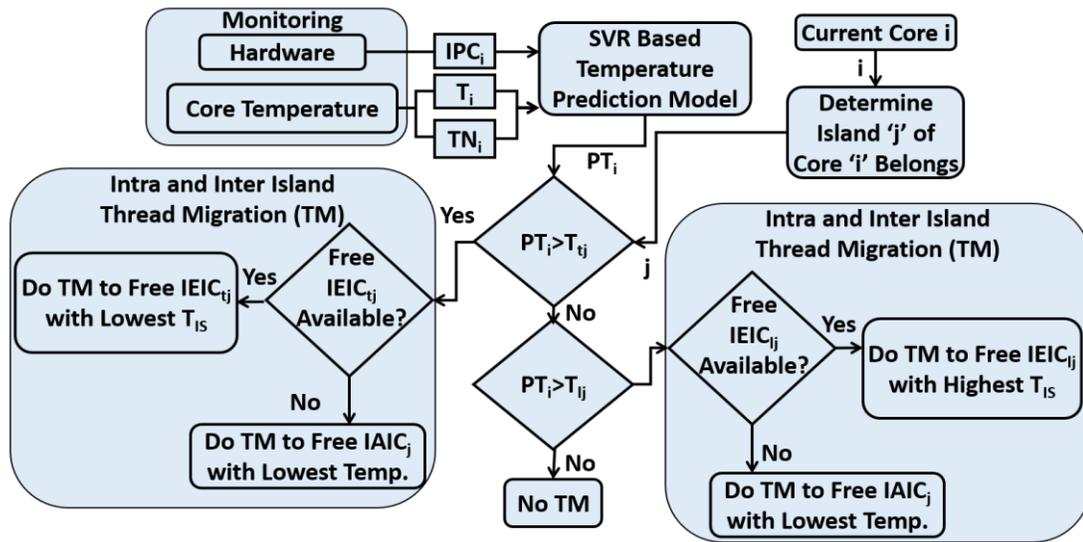


Figure 57 Overview of TATM technique with support vector regression (SVR) based temperature prediction model.

8.2.2.3. THERMAL MANAGEMENT ALGORITHM

Figure 57 illustrates the entire TATM technique. For each core, we periodically monitor the IPC value from performance counters and temperature from thermal sensors. If a thermal emergency is predicted for a core by the SVR predictor, then TATM initiates a thread migration procedure, otherwise no action is taken. In this chapter we have considered the thermal threshold of an island to be equal to maximum allowable temperature in that island i.e. $T_{tj} = T_{1Sj} + 10K$ to avoid instances of irrevocable drift in MRs and thermal limit of an island is minimum allowable temperature in that island i.e. $T_{lj} = T_{1Sj} - 10K$ to reduce tuning power.

Algorithm 3 TATM thread migration algorithm

Inputs: Current core temperature (T_i), average neighboring core temperature (TN_i), current core IPC (IPC_i)

```
1:  for each core i do // Loop that predicts future temperature
2:     $PT_i = SVR\_predict\_future\_temperature(T_i, TN_i, IPC_i)$ 
3:  for each core i do // Loop that checks for free IAICs
4:    j = Find island of core (i)
5:    if  $IPC_i == 0$  then List_IAICj = Push i //add core to IAICj list
6:  for each island j do // Loop that create IEIC list for  $T_{IS}$ -island
7:    for all islands m do
8:      if  $T_{ISM} > T_{ISj}$  then IEICij = push IAICm
9:      else if  $T_{ISM} < T_{ISj}$  then IEICij = push IAICm
10: for each core i do // Loop that performs thread migration (TM)
11:   j = Find island of core (i)
12:   if  $PT_i > T_{ij}$  then // Check predicted temp exceed thermal threshold
13:     if List_IEICij ≠ {} // Do inter-island TM
14:       Migrated_core = Find_lowest_ $T_{IS}$ _core(List_IEICij)
15:       Thread_migration(core_i → Migrated_core)
16:       n = island of Migrated_core
17:       List_IAICn and List_IEICij = Pop Migrated_core
18:     else if List_IAICj ≠ {} then // Do intra-island TM
19:       Migrated_core = Find_min_temp_core(List_IAICj)
20:       Thread_migration(core_i → Migrated_core)
21:       List_IAICj and List IEIC = Pop Migrated_core
22:   else if  $PT_i < T_{ij}$  then // if predicted temp is below thermal limit
23:     if List_IEICij ≠ {} // Do inter-island TM
24:       Migrated_core = Find_highest_ $T_{IS}$ _core(List_IEICij)
25:       Thread_migration(core_i → Migrated_core)
26:       n = island of Migrated_core
27:       List_IAICn and List_IEICij = Pop Migrated_core
28:     else if List_IAICj ≠ {} then // Do intra-island TM
29:       Migrated_core = Find_min_temp_core(List_IAICj)
30:       Thread_migration(core_i → Migrated_core)
31:       List_IAICj and List IEIC = Pop Migrated_core
```

Output: Thread migration to IAIC or IEIC cores

Algorithm 3 shows the pseudo-code for the TATM thread migration procedure. Firstly, future temperature (PT_i) of the i^{th} core is predicted using the SVR based predictor with inputs: core temperature (T_i), core IPC (IPC_i), and temperature of neighboring cores (TN_i) in steps 1-2. The list of available free cores ($IAIC_j$) in T_{ISj} -island (i.e., those that are not currently executing any thread) is obtained in steps 3-5. In steps 6-9, a loop iterates over islands to generate a list of free cores $IEIC_{ij}$ and $IEIC_{ij}$ in other islands whose T_{IS} is higher and lower than current island respectively. In step 10, a loop iterates over all cores to perform thread migration. Step 12 and 22 checks for

possible thread migration conditions (i.e., thermal emergency cases where current core predicted temperature (PT_i) in T_{ISj} -island is greater than thermal threshold (T_{ij}) or smaller than thermal limit (T_{lj})). If a thread migration is required as $PT_i > T_{ij}$, then in steps 13-21, we check for free $IEIC_{lj}$, and if they are available then we migrate the thread from the current core to the $IEIC_{lj}$ core with the lowest T_{IS} (inter-island migration), else we migrate the thread to a free $IAIC_j$ with the lowest temperature (intra-island migration). On the other hand, if a thread migration is required as $PT_i < T_{lj}$, then in steps 23-31, we check for free $IEIC_{lj}$, and if they are available then we migrate the thread from the current core to the $IEIC_{lj}$ core with the highest T_{IS} (inter-island migration), else we migrate the thread to a free $IAIC_j$ with the lowest temperature (intra-island migration). This TATM thread migration technique is invoked at every 1ms (epoch) and the sample frequency of SVR is considered as 0.1 ms (10 times lower compared to the epoch for thread migration). This sampling frequency is sufficient to monitor on-chip temperature variations [139].

8.3. EXPERIMENTS, RESULTS, AND ANALYSIS

8.3.1. EXPERIMENT SETUP

The IPKISS [140] tool was used for the design and simulation of heaters, MRs, and other silicon photonic components. This tool allows photonic component layout design, virtual fabrication of components in different technologies, physical simulation of components, and optical circuit design and simulation. The circuit-level results obtained from IPKISS were used for system-level simulation.

We target a 64-core CMP system for evaluation of our IHDTM framework. Each core has a Nehalem x86 [141] microarchitecture with 32 KB L1 instruction and data caches and a 256 KB L2 cache, at 32nm and running at 5GHz. We evaluate our framework on two well-known PNoC

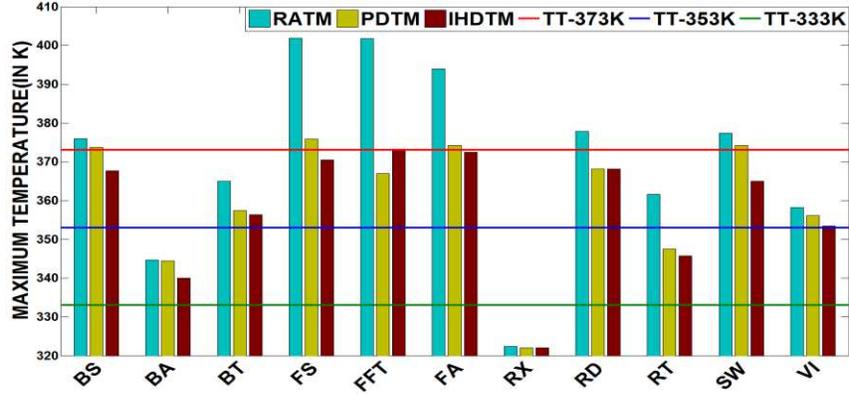
architectures: Corona [67] and Flexishare [13]. Corona uses a 64×64 multiple write single read (MWSR) crossbar with token slot arbitration. Flexishare uses 32 multiple write multiple read (MWMR) waveguide groups with a 2-pass token stream arbitration. Each MWSR waveguide in Corona and each MWMR waveguide in Flexishare is capable of transferring 512 bits of data from a source node to a destination node.

We modeled and simulated these architectures with the IHDTM framework for multi-threaded applications from the PARSEC [43] and SPLASH-2 [131] benchmark suites (Section 8.2.2). Simulations were performed with an execution period of one billion cycles. Power and instruction traces for the benchmark applications were generated using the Sniper 6.0 [141] simulator and McPAT [142]. We used the 3D-ICE tool [130] for thermal analysis. We considered a three layered 3D-stacked CMP system as advocated in existing PNoC architectures [11], [13] with a planar die area footprint of 400mm², where the top layer is the core-cache layer, the middle layer is the analog electronic layer [67] which contains control circuits for modulator and photodetector and also the trans-impedance amplifiers of detectors, and the bottom layer is the photonic layer with MRs, waveguides, ring heaters, and ring trimmers for carrier injection. Some of the key materials used in the construction of the 3D-stack in the 3D-ICE tool and their properties are shown in Table II. We used a heat sink adjacent to the core-cache layer for heat dissipation to the ambient environment.

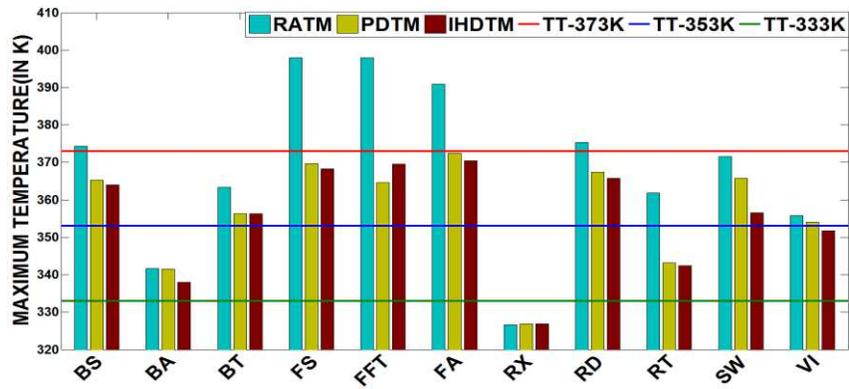
Table 20 Properties of materials used by 3D-ICE tool [130], [143]

Material	Thermal Conductivity	Volumetric Heat Capacity
Silicon	1.30e-4 W/μm K	1.628e-12 J/μm ³ K
Silicon dioxide	1.46e-6 W/μm K	1.628e-12 J/μm ³ K
BEOL	2.25e-6 W/μm K	2.175e-12 J/μm ³ K
Copper	5.85e-4 W/μm K	3.45e-12 J/μm ³ K

BEOL: Back end of line fabrication material



(a)



(b)

Figure 58 Maximum temperature comparison of IHDTM with RATM and PDTM for (a) 48 and (b) 32 threaded PARSEC and SPLASH-2 benchmarks executed on 64-core CMP with Corona PNoC.

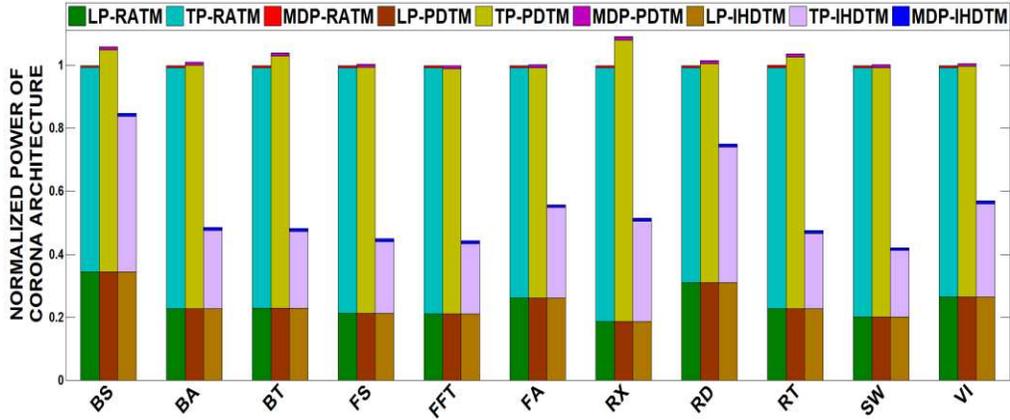
The MR thermal sensitivity was assumed to be 0.11nm/K [19]. For PNoCs, we considered 64 dense-wavelength-division-multiplexing (DWDM) waveguides sharing the working band 1530-1625 nm. The MR trimming power is set to 130 μ W/nm [18] for current injection (blue shift) and tuning power is set to 240 μ W/nm [19] for heating (red shift). To compute laser power, we considered detector responsivity as 0.8 A/W [26], MR through loss as 0.02 dB, waveguide propagation loss as 1 dB/cm, waveguide bending loss as 0.005 dB/90⁰, and waveguide coupler/splitter loss as 0.5 dB [26]. We calculated photonic loss in components using these values,

which sets the photonic laser power budget and correspondingly the electrical laser power. For energy consumption of photonic devices, we adapt parameters from [31], with 0.42pJ/bit for every modulation/detection, and 0.18pJ/bit for modulator/detector driver circuits.

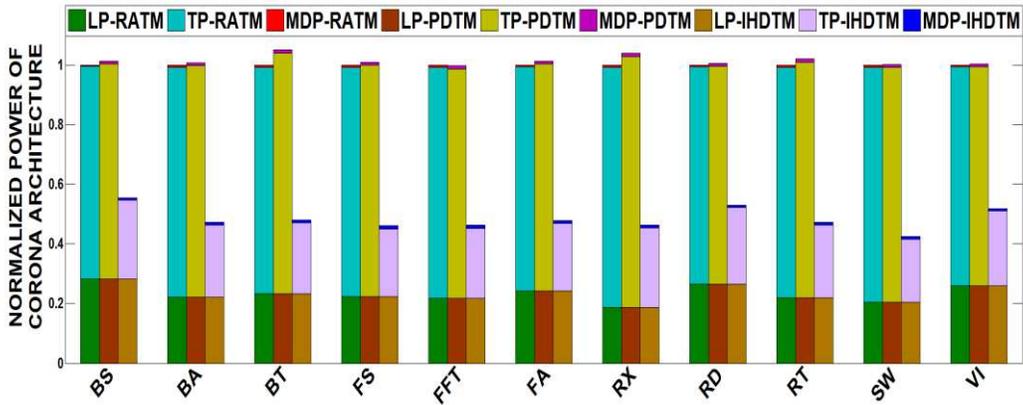
The ambient temperature was set to 303K for our analysis and the for $T_{IS1-island}$, $T_{IS2-island}$, and $T_{IS3-island}$ thermal thresholds were set to 373K, 353K, and 333K respectively and the thermal limits were set to 353K, 333K, and 313K respectively. Based on our sensitivity analysis we get the best accuracy for our SVR-based temperature predictor when parameters C and γ are set to 1000 and 0.1 respectively. We also considered thread migration overhead in our simulations that ranged from 500-1000 cycles to account for startup latency (extra cache misses, branch miss predictions) in the migrated core. Further, in the simulation we considered a 250-500 cycles overhead towards migration of threads for writing dirty cache lines from the write back caches, flushing the pipeline in the source core, and also PNoC latency to transfer data from architectural registers from the source core to the migrated core.

8.3.2. EXPERIMENTAL RESULTS

We compared the performance of our IHDTM framework with two prior works on multicore thermal management: a ring aware policy (RATM) [133] and a predictive dynamic thermal management (PDTM) framework [139]. To compare these frameworks, we consider Corona and Flexishare PNoC architectures. RATM distributes threads uniformly across cores that are closer to PNoC nodes first and then distributes the remaining threads in a regular pattern from outer cores to inner cores. PDTM uses a recursive least square based temperature predictor to determine if the predicted temperature of a core exceeds a thermal threshold, and if so then thread migration is performed from that core to the coolest free core.



(a)



(b)

Figure 59 Normalized power (Laser Power (LP), Trimming and tuning power (TP) and modulating and detecting Power (MDP)) comparison of IHDTM with RATM and PDTM for (a) 48 and (b) 32 threaded applications of PARSEC and SPLASH-2 suites executed on Corona PNoC architectures for a 64-core multicore system. Results shown are normalized w.r.t RATM.

Figure 58 shows the maximum temperature obtained with the three frameworks across eleven applications from the PARSEC and SPLASH-2 benchmarks suites with 48 and 32 thread counts executed on a 64-core system with the Corona PNoC architecture. From Figure 58(a) it can be observed that for the IHDTM framework the FFT application with 48 threads exceeds the threshold (363K) by 0.4K as there are insufficient number of free cores in the 363K-island on the chip whose temperature is below the thermal threshold to migrate threads. However, in Figure

58(b) our IHDTM framework avoids violating thermal thresholds for all the benchmark applications with 32 threads. On average, IHDTM has 13.27K and 13.72K lower maximum temperature compared to the RATM policy for 48 and 32 threads, respectively. Along with local thermal stabilization by PID controlled heaters, IHDTM migrates threads from hotter cores to cooler cores to control maximum temperature, whereas RATM does a simple thread allocation that is unable to appropriately control maximum temperature. For most of the cases, maximum temperatures with PDTM and IHDTM are below the thermal threshold. On average, IHDTM has 2.37K and 1.56K lower maximum temperature compared to the PDTM policy for 48 and 32 threads, respectively. IHDTM prefers to migrate threads within islands (inter-island) of cores based on the power consumption of running thread, which facilitates reduction in its peak temperature compared to PDTM.

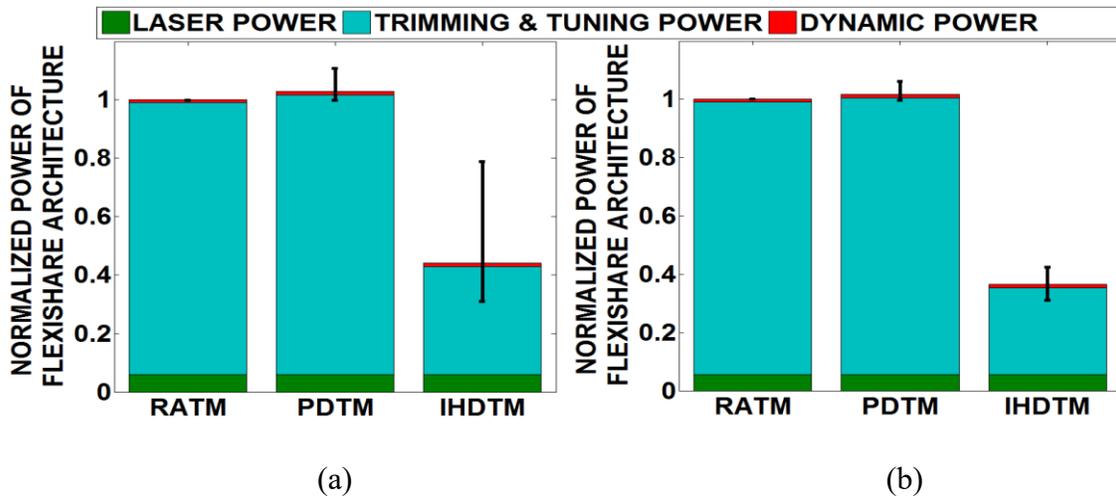
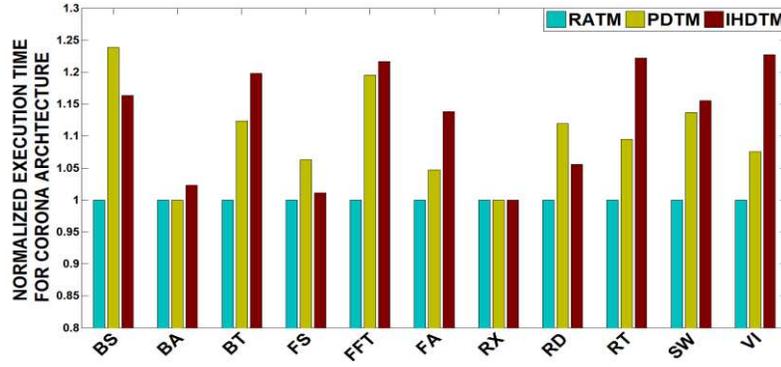
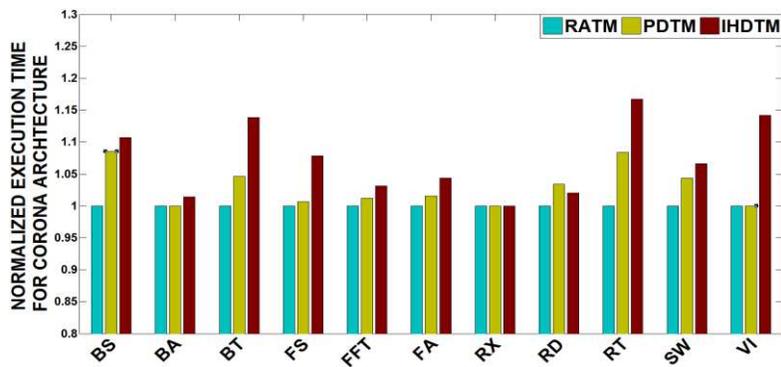


Figure 60 Normalized average power (laser power (LP), trimming and tuning power (TP) and modulating and detecting power (MDP)) comparison of IHDTM with RATM and PDTM for (a) 48 and (b) 32 threaded applications of PARSEC and SPLASH-2 suites executed on Flexishare PNoC for a 64-core system. Power results are normalized wrt RATM results. Bars represent mean values of power dissipation; confidence intervals show variation in power across PARSEC and SPLASH-2 benchmarks.



(a)



(b)

Figure 61 Normalized execution time comparison of IHDTM with RATM and PDTM for (a) 48 and (b) 32 threaded applications of PARSEC and SPLASH-2 suites executed on Corona PNoC for a 64-core system. Results shown are normalized w.r.t RATM.

IHDTM saves considerable thermal tuning and trimming power to ultimately reduce total power. From the power analysis in Figure 59 and Figure 60, it can be observed that IHDTM with Corona running 48 threads has 45.5% and 46.8%; and IHDTM with Corona running 32 threads has 51.6% and 52.3% lower total power consumption compared to Corona with RATM and PDTM respectively. Further, Flexishare with IHDTM running 48 threads has 55.9% and 57.2%; and 32 threads has 63.5% and 64.1% lower power consumption compared to Flexishare with RATM and PDTM respectively.

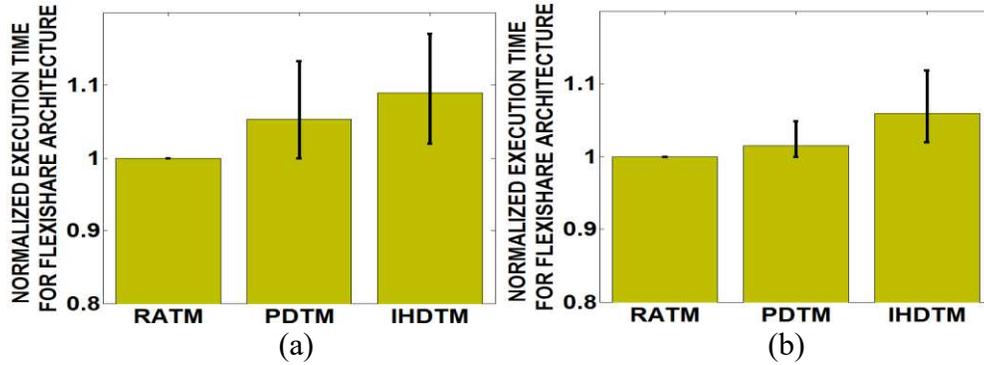


Figure 62 Normalized average execution time comparison of IHDTM with RATM and PDTM for Flexishare PNoC running (a) 48; and (b) 32 threaded applications from PARSEC and SPLASH-2 suites executed on 64-core system. Results are normalized wrt RATM results. Bars represent mean values of execution time; confidence intervals show variation in execution time across PARSEC and SPLASH-2 benchmarks.

Figure 61 shows the average execution time comparison between the three frameworks across the 11 48-threaded and 32-threaded applications from the PARSEC and SPLASH-2 suites, for the Corona PNoC architectures respectively. From Figure 61(a) and (b) it can be seen that Corona with IHDTM running 48 and 32 threads has 12.8% and 7.4% higher execution time respectively compared to Corona with RATM. Corona with IHDTM needs extra execution time to migrate threads between cores whereas the RATM policy simply schedules threads without any migration, and thus does not possess such overheads. The execution time overhead of Corona with IHDTM running 32 threads is lower compared to 48-threaded version, as it lowers traffic congestion in the Corona PNoC which in turn reduces overall latency. Further, Corona with IHDTM running 48 and 32 threads has 2.6% and 4.3% higher execution time respectively compared to PDTM. IHDTM has more number of thread migrations compared to the number of thread migrations in PDTM, as IHDTM performs intra-island and inter-island thread migrations when the thermal emergencies are predicted by the SVR predictor. Similarly, from Figure 62(a) and (b), the Flexishare with IHDTM running 48 and 32 threads has 9% and 5.9% higher execution time compared to RATM and 3.4% and 4.4% higher execution time compared to Flexishare with

PDTM. From the execution time results it can be seen that Flexishare has lower execution time overhead compared to Corona as it uses a faster MWMR crossbar instead of slower MWSR crossbar in Corona.

Lastly, from the power consumption and execution time results, we can obtain energy consumption results for the three frameworks. On an average, for Corona, energy consumption of IHDTM running 48 threads is 38.5% and 45.4% lower compared to RATM and PDTM, respectively. Further energy consumption of Corona with IHDTM running 32 threads is 48.1% and 50.3% lower compared to RATM and PDTM, respectively. On the Flexishare architecture, IHDTM running 48 threads has 52.2% and 56% lower energy consumption compared to RATM and PDTM respectively; and IHDTM running 32 threads has 61.4% and 62.6% lower energy consumption compared to RATM and PDTM, respectively. From the energy consumption results IHDTM has better energy savings for the optimized Flexishare compared to the Corona.

8.4. CONCLUSIONS

We have presented the IHDTM framework that exploits device-level on-chip thermal islands and system-level dynamic thread migration scheme TATM for the reduction of maximum on-chip temperature and also conserves trimming and tuning power of MRs in DWDM-based PNoC architectures. The proactive thermal management scheme used in IHDTM results in interesting trade-offs between performance and power/energy across two different state-of-the-art crossbar-based PNoC architectures. Our experimental analysis on the well-known Corona and Flexishare PNoC architectures has shown that IHDTM can notably conserve total power by up to 64.1% and thermal tuning power by up to 70%.

9. LIBRA: THERMAL AND PROCESS VARIATION AWARE RELIABILITY MANAGEMENT IN PHOTONIC NETWORKS-ON-CHIP

PNoCs operation is very sensitive to on-chip temperature and process variations. These variations can create significant reliability issues for PNoCs. For example, a microring resonator (MR) may resonate at another wavelength instead of its designated wavelength due to thermal and/or process variations, which can lead to bandwidth wastage and data corruption in PNoCs. This chapter proposes a novel run-time framework called *LIBRA* to overcome temperature- and process variation- induced reliability issues in PNoCs. The framework consists of (i) a device-level reactive MR assignment mechanism that dynamically assigns a group of MRs to reliably modulate/receive data in a waveguide based on the chip thermal and process variation characteristics; and (ii) a system-level proactive thread migration technique to avoid on-chip thermal threshold violations and reduce MR tuning/ trimming power by dynamically migrating threads between cores. Our simulation results indicate that *LIBRA* can reliably satisfy on-chip thermal thresholds and maintain high network bandwidth while reducing total power by up to 61.3%, and thermal tuning/trimming power by up to 76.2% over state-of-the-art thermal and process variation aware solutions.

9.1. INTRODUCTION

As advocated by prior works [11], [12], PNoCs are expected to be 3D-stacked on top of their respective manycore chips. Therefore, the MRs of PNoCs will be placed on top of, and hence in close proximity to, processing cores. Variations in core workloads lead to variations in their power dissipation, which in turn can alter the temperatures of the cores and MRs in their vicinity. For

instance, the temperature on a typical manycore chip can easily vary by as much as 90⁰C [144]. Unfortunately, MRs are very sensitive to these on-chip thermal variations (TV): their effective refractive indices, and hence their resonance wavelengths are altered if their operating temperatures change. Therefore, in a typical PNoC, the resonance wavelengths of the utilized modulator MRs may not align with, and hence may not modulate their assigned carrier wavelengths [19]. This may result in bandwidth wastage, or worse, data corruption when detector MRs are unable to read from their assigned carrier wavelengths [33].

In addition to TV, MRs are also susceptible to fabrication process variations. Process variations (PV) induce variations in the width, thickness, and doping concentration of MRs (see Section 9.3.2), causing resonance wavelength shifts in MRs [21], [22]. PV measurements of fabricated MR devices indicate a standard deviation (σ) of 1.3 nm in width, which translates to a 0.76nm shift in an MR's resonance wavelength [23]. These PV-induced resonance wavelength shifts in MRs also cause bandwidth wastage and data corruption.

The adverse effects of PV and TV related to resonance shifts in MRs, and their performance and reliability impacts, can be redressed by realigning the resonant wavelengths of MRs with their assigned carrier wavelengths using localized trimming [18] and thermal tuning [19] mechanisms. Trimming alters the free-carrier concentration in an MR core, whereas thermal tuning uses integrated micro-heaters to alter local temperatures at MRs. But these mechanisms come with high power and performance overhead [19]. *Hence, it is essential to intelligently manage thermal and process variations in PNoC-based manycore systems, to achieve reliable communication with minimal trimming and tuning costs.*

In this chapter, we aim to minimize the need for (and overheads of) localized thermal tuning and trimming in PNoCs while coping with process and thermal variations, thereby easing the

adoption of PNoCs in future manycore systems. We propose a novel thermal and process variation aware dynamic reliability management framework called *LIBRA* that integrates adaptive MR assignment at the device-level and dynamic thread migration at the system-level for PNoC-based manycore systems. Our novel contributions as part of the *LIBRA* framework are summarized below:

- We design a novel thermal and process variation aware MR assignment (*TPMA*) mechanism at the device-level, which dynamically assigns a set of MRs to the utilized set of carrier wavelengths at run-time. *TPMA* enables reliable modulation and reception of data with minimal overheads, while maintaining the maximum possible bandwidth;
- We propose a novel PV-aware anti wavelength-shift dynamic thermal management (*VADTM*) mechanism at the system-level, which uses support vector regression (*SVR*) based temperature prediction (see section 8.2.2.2 of chapter 8) and dynamic thread migration to avoid on-chip thermal threshold violations and reduce trimming/tuning power for MRs;
- We evaluate our *LIBRA* (*TPMA+VADTM*) framework on a 64-core chip, and compare it with four state-of-the-art thermal management solutions: an MR-aware thermal management (*RATM*) framework [133], an MR PV-aware thermal management (*FATM*) framework [145], a predictive dynamic thermal management (*PDTM*) framework [139], and an MR-aware thermal management (*SPECTRA*) framework [33]; and show significant reduction in maximum temperature and trimming/tuning power costs compared to these solutions.

9.2. RELATED WORK

Traditional electrical NoC communication fabrics are projected to suffer from cripplingly high power dissipation and severely reduced performance in future manycore systems [7]. The

higher bandwidth density and lower power dissipation possible with silicon-photonic links, compared to electrical wires, has made them an attractive option for manycore systems. Recent research has thus focused on exploring a wide spectrum of network topologies and protocols to enable efficient PNoC architectures [25], [67].

PV and TV in silicon-photonic links represent important challenges for the widespread adoption of PNoC architectures. Several techniques have been proposed to reduce thermal hotspots and gradients using DVFS [146], [147], [148], [149], workload migration [139], [150], [151] and liquid cooling [152], [153], [154], [155]. A few PV-aware application mapping frameworks have also been proposed [156], [157] that optimize performance and energy in manycore systems. In [156] a run-time application-mapping strategy was presented, which considers the variation profile of a manycore processor to maximize performance and reduce leakage-power for a given fixed power budget. In [157] a framework was presented that integrates reliability and variation-awareness in a run-time variable degree-of-parallelism (DoP) application-scheduling methodology to enhance manycore performance. *However, these techniques do not consider the unique challenges (e.g., MR resonance wavelength shifts) and constraints (e.g., wavelength match between sender and receiver MR pairs) that exist in PNoCs.*

A few prior works have analyzed the impact of TV and PV on PNoCs at the device-level, link-level, and system-level, and proposed solutions to remedy these variations. The device-level efforts have mainly proposed various athermal photonic devices to reduce localized tuning/trimming power in MRs. These design-time solutions include using materials such as cladding to reduce thermal sensitivity [132], [158], and using heaters and temperature sensors for thermal control [159]. An electrical backend capable of bit re-shuffling was proposed in [160] to enhance photonic link robustness against TV and PV with lower MR tuning power. *While these*

device- and link-level techniques are promising, they either possess a high power overhead or require costly changes in the manufacturing process (e.g., larger device areas) that would decrease bandwidth density and area efficiency.

At the system-level, the overhead associated with localized tuning of MRs was reduced in [19] using the group shift property of co-located MRs as part of a method to trim a group of rings at the same time. In [133], a ring-aware thread scheduling policy was proposed to reduce on-chip thermal gradients in a PNoC. In [161], a thread migration mechanism was proposed to minimize on-chip thermal gradients within a PNoC. In [34], an island of heater based thermal management framework was proposed to adapt groups or islands of MRs within PNoCs to on-chip thermal variations. A few prior works have also explored the impact of PV on DWDM-based photonic links at the system-level [105], [162], [163]. A reliability-aware design flow to address variation induced reliability issues is proposed in [162], which uses athermal coating at fabrication-level, voltage tuning at device-level, as well as channel hopping at the system-level. In [105], a methodology to salvage network-bandwidth loss due to PV-shifts is proposed, which reorders MRs and trims them to nearby wavelengths. In [163], power-efficient techniques are proposed, based on inter-channel hopping and variation-aware routing to compensate for PV effects at runtime. A few system-level works [23], [145], [164] also consider the impact of both TV and PV on optical links. In [23], a thermal-tuning approach is presented that adjusts chip temperature using DVFS to compensate for chip-wide thermal and process variation induced resonance shifts in MRs and improve system performance. In [145], a PV aware workload allocation policy is presented to reduce the thermal tuning power of PNoCs. In [164], a tunable laser source design is demonstrated, in which the signal power at the source is adapted to compensate for signal losses due to TV and

PV across optical interconnects. *None of these system-level solutions for PNoCs considers the impact of the relationship between thermal hotspots and transmission reliability.*

To address these shortcomings of prior work, we proposed the SPECTRA framework in our prior work [33]. SPECTRA is a cross-layer framework that combines two dynamic thermal management mechanisms to reduce maximum on-chip temperature and conserve trimming and tuning power of MRs in DWDM-based PNoC architectures. Our proposed *LIBRA* framework in this chapter improves upon SPECTRA, by (i) considering the impact of PV on dynamic thermal management; (ii) utilizing a new device-level TV and PV aware ring assignment mechanism; and (iii) utilizing a new system-level PV-aware thread migration mechanism. Sections 4-6 describe our proposed framework which is then evaluated in Section 7 against prior work.

9.3. IMPACT OF TV AND PV ON DWDM BASED PNOCS

In this section, we explain the key impacts of PV and TV on DWDM based PNoCs. Although most silicon-based photonic devices exhibit some susceptibility to temperature and process variations, the high wavelength selectivity of MRs makes them especially susceptible to these variations. Therefore, we primarily focus on the impacts of TV and PV on MRs.

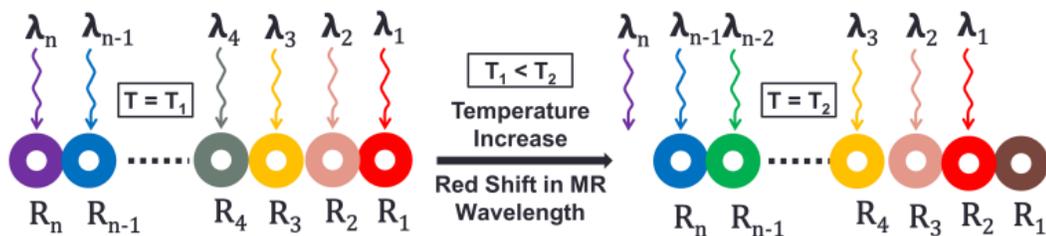


Figure 63 Impact of temperature increase on an MR bank

9.3.1. IMPACT OF TV ON DWDM BASED PNOCS

In a DWDM PNoC, the temperatures of the individual compute nodes and their associated MR banks follow the workload-dependent temperatures of the processing cores in the nodes. As the application workload of each core in a manycore system usually differs from that of other cores and also varies with time, the temperatures of the cores (and thus nodes) of the system differ from one-another and vary with time. As a result, the temperatures of different MR banks of the PNoC also differ from one another and vary with time.

Typically, the MR banks of each PNoC node are designed to resonate with and operate upon their assigned carrier wavelengths at a specific temperature, e.g., room temperature. But due to the time- and workload-dependent temperature variations, the resonances of different MR banks shift away from their assigned carrier wavelengths by different amounts.

For example, Figure 63 depicts an MR bank with MRs R_1 - R_n that are designed to resonate with their assigned carrier wavelengths λ_1 - λ_n , respectively, at temperature T_1 . As the temperature increases to T_2 , the resonance wavelength of each MR shifts away from its assigned carrier wavelength towards the red end of the spectrum (i.e., *red-shift*). This red-shift is shown in the figure where, at temperature T_2 ($T_2 > T_1$), the resonance wavelength λ_i of MR R_i is in line with the carrier wavelength λ_{i-1} . Consequently, the carrier wavelength λ_n is not assigned to any of the MRs. This results in *bandwidth wastage* if the MR bank is a modulator MR bank, as λ_n cannot be modulated by any modulator MR now. This example scenario can also result in *data corruption* if the MR bank is a detector MR bank, as λ_n cannot be received by any detector MR. Similarly, if $T_2 < T_1$, the resonance wavelength λ_i of each R_i shifts towards the blue end of the spectrum (i.e., *blue-shift*), which may leave λ_i unassigned, causing bandwidth wastage or data corruption. Thus, during

the runtime of a PNoC, an increase in an MR bank's temperature red-shifts the resonances of all its MRs, whereas a decrease in an MR's temperature blue-shifts the resonances of all its MRs.

The amount of shift in an MR's resonance not only depends on the magnitude of temperature change, but also on the MR's structure and geometry manifested as its effective refractive index n_{eff} . Typically, an MR is a looped waveguide with a silicon (Si) core and silicon dioxide (SiO₂) cladding, irrespective of whether it is used as a modulator or a detector. The change $\Delta\lambda_r$ in the resonance wavelength λ_r of an MR due to an arbitrary change ΔT in its local temperature is given by the following equation [113]:

$$\frac{\Delta\lambda_r}{\Delta T} = \frac{\delta n_{eff}}{\delta T} \frac{\lambda_r}{n_g} = \left(\Gamma_{Si} \frac{\delta n_{Si}}{\delta T} + \Gamma_{SiO_2} \frac{\delta n_{SiO_2}}{\delta T} \right) \frac{\lambda_r}{n_g}, \quad (61)$$

Here, n_g is the group refractive index (ratio of speed of light to group velocity of all wavelengths traversing the waveguide) of the MR waveguide. Γ_{Si} and Γ_{SiO_2} are the modal confinement factors of the MR's core (Si) and cladding (SiO₂), respectively. $\delta n_{Si}/\delta T$ and $\delta n_{SiO_2}/\delta T$ are the thermo-optic coefficients of Si (MR's core) and SiO₂ (MR's cladding) materials, with values of $1.86 \times 10^{-4} \text{ K}^{-1}$ and $1 \times 10^{-5} \text{ K}^{-1}$, respectively [113]. As the thermo-optic coefficient of Si is an order of magnitude greater than that of SiO₂, and as Γ_{Si} is also greater than Γ_{SiO_2} for a typical MR, the contributions from the MR's cladding (SiO₂) in Eq. (61) can be ignored. Consequently, Eq. (61) reduces to:

$$\Delta\lambda_r = \Gamma_{Si} \cdot \frac{\delta n_{Si}}{\delta T} \cdot \frac{\lambda_r}{n_g} \cdot \Delta T, \quad (62)$$

Note that the MRs used in this study are looped channel waveguides with a cross section of 450nm×220nm. We model these MRs using a commercial-grade eigenmode solver [122], based on which the values of Γ_{Si} and n_g at 1550nm are calculated to be 0.78 and 4.16, respectively.

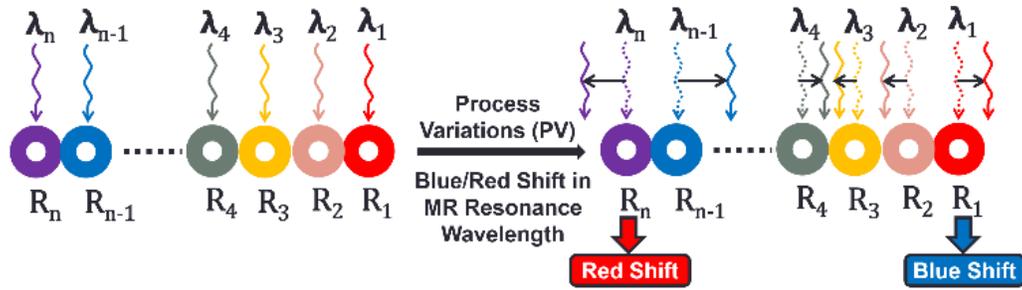


Figure 64 Impact of PV on DWDM based PNoCs

9.3.2. IMPACT OF PV ON DWDM BASED PNOCS

Ideally, without any fabrication-induced PV, a sender or a receiver node can modulate and detect all of the carrier wavelengths available in the waveguide without any bandwidth loss or error. But in reality, similar to deep submicron electronic devices, photonic devices such as MR modulators, MR detectors, grating couplers, splitters etc. also suffer from significant PV [165]. In this chapter, we mainly focus on the severe PV effects in MRs. The MR structure is very sensitive to PV, much like it is to TV. Due to PV effects, the widths, heights, and side wall roughness of MRs can deviate from desired values after fabrication. Consequently, the resonance wavelengths (λ_r) of the MRs also deviate from their designed values. For example, 1nm of variation in width and height of an MR can lead to 0.58~1nm and ~2nm shift in its resonance wavelength, respectively [22].

As discussed earlier, PNoCs employ DWDM-based photonic links with cascaded MRs (i.e., MR banks) in their sending and receiving nodes. Unlike TV that induces systematic red or blue shifts in all the MRs of an MR bank, PV can incur random shifts in the resonance wavelengths of the MRs of a single bank, as shown in Figure 64. From this figure, MRs R_1, R_4, \dots, R_{n-1} have blue shift in their resonance wavelengths and MRs R_2, R_3, \dots, R_n have red shift in their resonance wavelengths. Much like with TV, PV can also throw the resonances of the MRs out of alignment

with their assigned carrier wavelengths, which can ultimately lead to bandwidth wastage and/or data corruption.

In summary, to enable reliable photonic communication, there is a need to mitigate the combined impact of TV and PV on PNoCs. This chapter presents a cross-layer framework that uses device-level and system-level enhancements to remedy the combined impact of TV and PV. Before discussing our proposed framework, we present our performance, power, and thermal setup for modeling manycore systems with PNoCs in the next subsection. We also present a characterization of the impact of TV and PV on the MRs of a typical DWDM PNoC based manycore system, in this next subsection.

9.3.3. MODELING TV AND PV IN PNO C ARCHITECTURES

To model and characterize TV and PV in a manycore system with a PNoC, we developed a simulation framework, which integrates performance, power, thermal, and variation simulators, as shown in Figure 65. We considered a three layered 3D-stacked 64-core system as advocated in existing PNoC architectures [11], [12] with a planar die area footprint of 400mm². The top layer is the core-cache layer, the middle layer is the conversion layer with digital and analog circuits that support electrical-to-optical (E/O) and optical-to-electrical (O/E) conversion of data, and the bottom layer is the photonic layer with photonic components and devices (e.g., MRs, waveguides, ring heaters, etc.) that comprise a PNoC.

We use Sniper [141] to simulate the performance of the manycore system while it executes multithreaded applications from SPLASH-2 [131] and PARSEC [43] benchmark suites. To factor in the varying system utilizations as a contributor to the dynamic TV in the processing cores and its impact on the associated photonic devices (e.g., MRs), we run each application on a target 64-

core system (see Section 9.7) with 8, 16, 32, 48, and 64 threads. To capture runtime behavior of an application, we generate performance traces using Sniper, which are fed to McPAT [142] to generate power traces at core-level granularity. We use published power dissipation data from Intel’s Single-Chip Cloud Computer (SCC), scaled to 32 nm, to calibrate our dynamic power data. The power traces generated by McPAT are given as inputs to the 3D-ICE tool [130] for transient thermal simulations (see Figure 65). Some of the key materials used in the construction of the 3D-stack in the 3D-ICE tool and their properties are shown in Table 21. Additionally, we consider a heat sink adjacent to the core-cache layer for dissipation of heat in the environment.

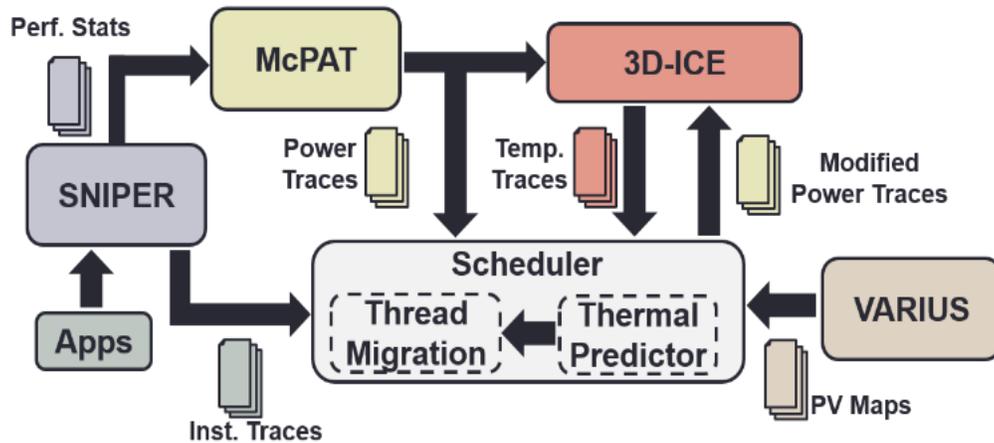


Figure 65 Simulation framework to analyze TV and PV in a manycore system with a PNoC architectures; the framework integrates performance, power, thermal, and variation simulators.

Table 21 Properties of materials used by 3D-ICE Tool [130], [143]

Material	Thermal Conductivity	Volumetric Heat Capacity
Silicon	1.30e-4 W/μm K	1.628e-12 J/μm ³ K
Silicon di oxide	1.46e-6 W/μm K	1.628e-12 J/μm ³ K
BEOL	2.25e-6 W/μm K	2.175e-12 J/μm ³ K
Copper	5.85e-4 W/μm K	3.45e-12 J/μm ³ K

We analyzed the spatial variation in the peak temperatures of various tiles (at core-level granularity) of the photonic layer. For the 64-core system each tile has an estimated area of 6.25 mm^2 (i.e. $2.5 \times 2.5 \text{ mm}^2$). We executed 64-threaded versions of the blackscholes (*BS*), bodytrack (*BT*), vips (*VI*), facesim (*FS*), fluidanimate (*FA*), swaptions (*SW*), barnes (*BA*), fft (*FFT*), radix (*RX*), radiosity (*RD*), and raytrace (*RT*) applications from the PARSEC and SPLASH2 benchmark suites on the 64-core system with one application running at a time. We monitored the peak temperature of each part of the photonic layer for every application and plotted the maximum peak temperature of each part across all the applications, as shown in Figure 66(a). From this figure, we can observe the maximum possible temperature-rise (above the room temperature) for any part of the layer, which caps all possible dynamic TV values for that part. From Fig. 4(a), higher peak temperatures are obtained at the center of the chip while relatively lower peak temperatures are achieved at the periphery of the chip. The main reason for the higher temperature at the center of the chip is the inefficiency of the heat sink to remove heat from the center of the chip. Furthermore, using Eq. (61) and (62), we determined the resonance wavelength shifts because of the peak temperature-rises, which are presented as a histogram in Figure 66(b). As evident from this figure, TV can induce up to a 7.4nm shift in MR resonances.

In addition to TV, we also analyzed PV in PNoCs with the simulation setup presented in Figure 65. We adapted the VARIUS tool [112] to model die-to-die (D2D) as well as within-die (WID) process variations in MRs for the PNoC. VARIUS uses a normal distribution to characterize on-chip D2D and WID process variations. The key parameters are mean (μ), variance (σ^2), and density (α) of a variable that follows the normal distribution. As wavelength variations are approximately linear to the dimension variations of MRs, we assume they follow the same distribution. The mean (μ) of wavelength variation of an MR is its nominal resonance wavelength.

For PNoCs, we considered waveguides with 32 DWDM degree sharing the working band 1530–1625nm (i.e., C and L bands) with a wavelength channel spacing of 1.48nm. Hence, those wavelengths are the means for each MR modeled. The variance (σ^2) of wavelength variation is determined based on laboratory fabrication data [22] and our target die size. For a 64-core chip with 400mm² size at 32nm node, we consider a WID and D2D standard deviations of $\sigma_{WID} = 0.61\text{nm}$ $\sigma_{D2D} = 1.01\text{nm}$, respectively [105]. We also consider a density (α) of 0.5 [105] for this die size. With these parameters, we use VARIUS to generate 100 process variation maps.

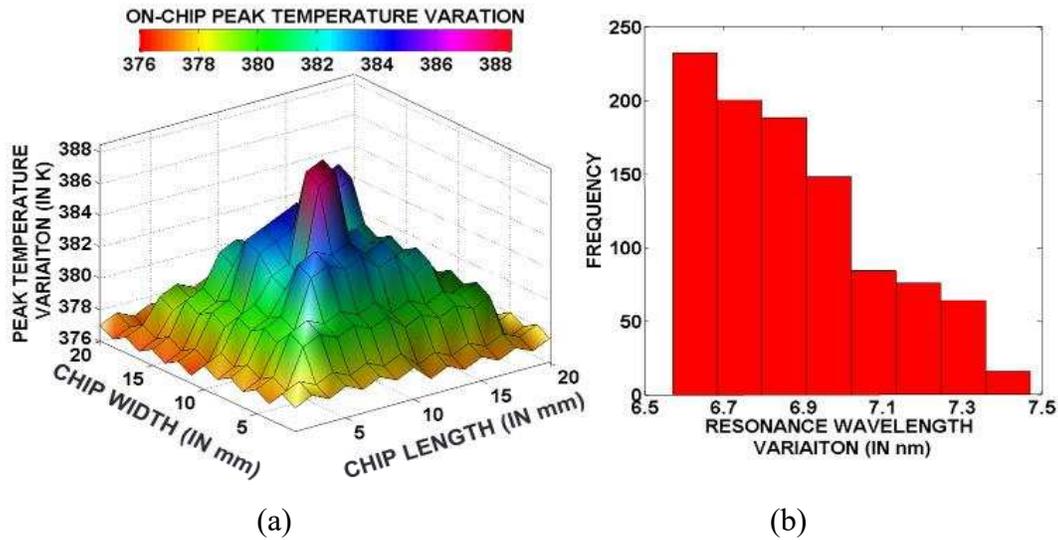


Figure 66 (a) spatial variation in peak temperatures (b) histogram of peak TV-induced resonance wavelength variation across a chip of size 400mm² using 3D ICE tool while executing 64 threaded PARSEC and SPLASH2 benchmark applications on a 64-core CMP.

We depict a PV map in Figure 67(a), which shows a spatial variation in PV-induced resonance wavelength shifts on the photonic die. Each PV map contains over one million points indicating the PV-induced shifts in MR resonances. The total number of points picked from these maps equal the number of MRs in the PNoC. We also present these points as a histogram in Fig. 5(b). As evident from the histogram, PV can induce resonance wavelength shifts in the range of -

1.8nm to 1.6nm. However, we observed that this range can increase up to -3nm to 3nm for other PV maps.

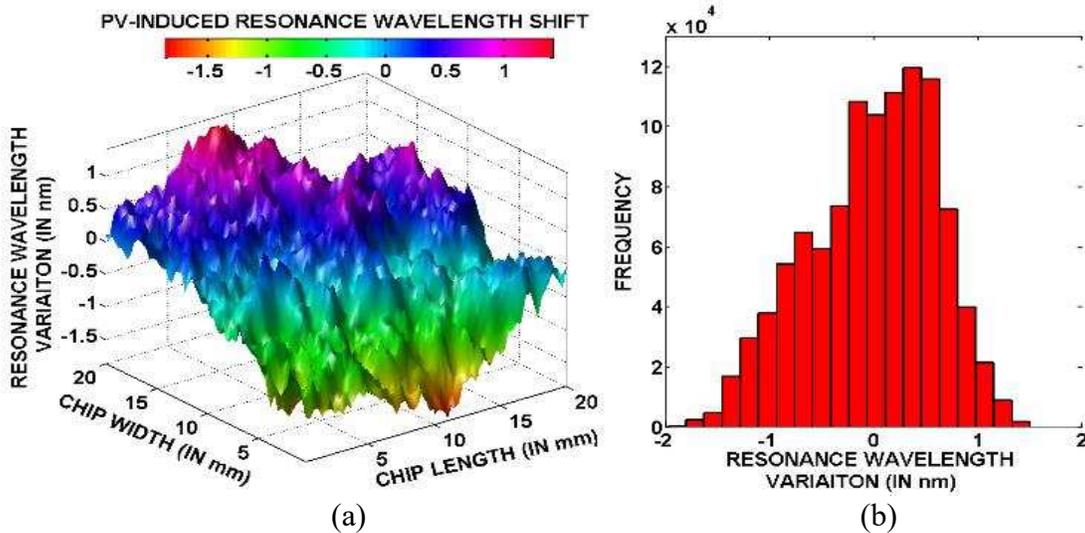


Figure 67 (a) PV-induced resonance wavelength variation (b) histogram of resonance wavelength variation across a chip of size 400 mm².

9.4. OVERCOMING PV/TV INDUCED RESONANCE WAVELENGTH SHIFTS

The adverse effects of PV and TV, i.e., resonance shifts in MRs and their performance and reliability impacts, can be overcome by realigning and locking the resonance wavelengths of the individual MRs with the utilized carrier wavelengths. As PV is a static phenomenon, the PV-induced resonance shifts need to be overcome only once at system initialization. In contrast, due to the dynamic nature of TV, the TV-induced resonance shifts require runtime thermal stabilization of MRs. A stable locking of MR resonances with the utilized carrier wavelengths can be achieved using device-level (MR-level) mechanisms, such as localized trimming [18] and/or thermal tuning [19], with a dithering signal based feedback control [113]. However, the localized trimming and thermal tuning mechanisms proposed in prior work come with several challenges, which must be overcome to ease the adoption of PNoCs for future manycore systems.

First, thermal tuning and localized trimming mechanisms cannot provide sufficient tuning range to remedy PV/TV-induced resonance shifts in MRs. For instance, from Section 9.3, TV and PV together can induce shifts in MR resonance wavelengths of up to 10.4nm, i.e., 7.4nm for TV and ± 3 nm for PV. Therefore, compensating these TV/PV-induced resonance shifts would require a net tuning range of 10.4nm. But localized trimming can provide a tuning range of only 1.5nm at most [162]. In contrast, thermal tuning can provide a tuning range of about 6.6nm corresponding to the temperature range of up to 60K [113] at 0.11nm/K sensitivity [19]. Thus, even the thermal tuning and localized trimming together (i.e., 6.6nm+1.5nm tuning range) cannot provide the required tuning range of ~ 10.4 nm. Another challenge for these mechanisms is their significant power overhead. A typical MR may consume 130 μ W of trimming power or 240 μ W of thermal tuning power to remedy 1nm shift in its resonance wavelength, depending on its size, structure, and integration feasibility. To remedy a larger shift of ~ 10.4 nm, a single MR may consume as much as ~ 1.35 mW of trimming power or ~ 2.5 mW of thermal tuning power. As a DWDM PNoC may have thousands of MRs, the total power overhead of PV/TV remedy can easily be in the range of a few tens of watts, which is a prohibitively high power overhead for chip-scale systems and must be minimized to make the total power costs of large-scale DWDM PNoCs manageable.

Fortunately, due to the periodicity of MR resonances, the resonance of none of the MRs in a PNoC needs to be tuned for more than a single channel gap [160]. This makes the required tuning range and the total tuning power more manageable. To understand this, consider Fig. 6. The periodic resonances (R_1 - R_4) of an example bank of four MRs and their assigned carrier wavelengths (λ_1 - λ_4) for an ideal case with no PV or TV are shown in Figure 68(a). Due to the absence of PV/TV, the resonances of all MRs are aligned with their assigned carrier wavelengths. Figure 68(b) shows systematic blue-shifts of over two channel gaps in the resonances of all four

MRs. In this case, the MR resonances can be re-aligned to their nearest carrier wavelengths followed by electrical repositioning of bits using backend barrel-shifters or pipelined shift registers [160]. In case the random PV throw the MR resonances out of order (Figure 68(c)), use of bit reordering multiplexers at the backend can still allow the MR resonances to be re-aligned to their nearest carrier wavelengths. Thus, due to the periodicity of MR resonances, and the use of bit reordering/repositioning techniques, the necessary tuning distance for the individual MRs reduces to less than one channel gap.

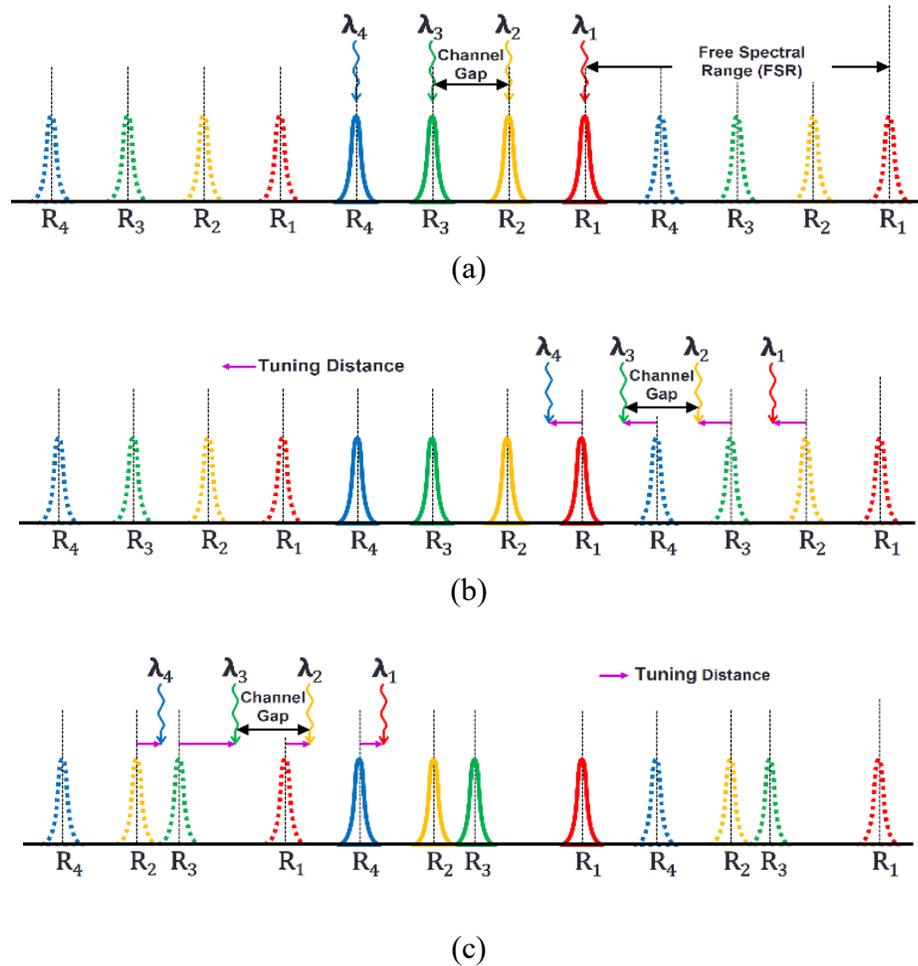


Figure 68 Periodic resonances (R_1 - R_4) of an example bank of four MRs and their assigned carrier wavelengths (λ_1 - λ_4) for (a) an ideal case with no resonance shifts, (b) a case with systematic blue-shifts in resonances, (c) a case with random red-shifts in resonances.

Our previously proposed SPECTRA framework [33] uses a different approach to reduce the required tuning distance and power overhead of PV/TV remedy. It integrates one system-level and two device-level optimizations. At the device-level, the SPECTRA framework utilizes three more MRs than the number of utilized carrier wavelengths, and thus, increases the available tuning range by three channel gaps. This mechanism reassigns the extra MRs to operate on nearby carrier wavelengths in the case when the resonances shift by less than three channel gaps. The need for remedying resonance shifts of more than three channel gaps is eliminated by reducing the range of temperature swings of the individual cores below the threshold levels that can induce resonance shifts of greater than three channel gaps. For that, an adaptive thread migration policy is used at the system level, which also eliminates the need of bit-shifting. Moreover, SPECTRA adaptively chooses the least power-consuming method from *thermal tuning* and *localized trimming* as the preferred method for PV/TV remedy. Thus, SPECTRA conserves the total power required for PV/TV remedy with low latency overhead. However, the SPECTRA framework does not deal with PV and its benefits come with the area and power overheads of the extra MRs and bit-reordering multiplexers [160]. *To address these shortcomings of the SPECTRA framework, we propose a new TV and PV aware reliability management framework called LIBRA, which is described next.*

9.5. LIBRA FRAMEWORK: OVERVIEW

Our *LIBRA* framework enables reliability-aware run-time PNoC management while rectifying TV and PV in MRs by integrating device-level and system-level enhancements. Figure 69 gives a high-level overview of our framework. The thermal and process variation aware microring assignment (*TPMA*) mechanism dynamically assigns each MR to the nearest available carrier wavelength, which enables reliable modulation and reception of data while maintaining the

maximum possible bandwidth. This device-level mechanism also adaptively chooses the least power-consuming method from thermal tuning and localized trimming as the preferred method for PV/TV remedy, and thus, reduces the total power for PV/TV remedy in the PNoC. However, limiting the peak temperature swings below threshold levels is critical to further reduce the total power for PV/TV remedy. To achieve this, we devise a PV-aware anti-wavelength-shift dynamic thermal management (*VADTM*) scheme that uses support vector regression (SVR) based temperature prediction and dynamic thread migration, to avoid on-chip thermal threshold violations, minimize on-chip thermal hotspots, and reduce thermal tuning power for MRs. The next two sections present details of the *TPMA* and *VADTM* schemes.

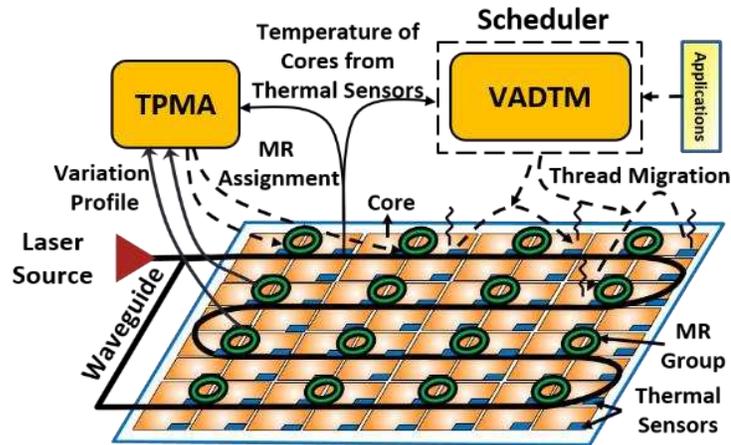


Figure 69 Overview of LIBRA framework that integrates a device-level thermal and process variation aware microring assignment mechanism (*TPMA*) and a system-level variation aware anti wavelength-shift dynamic thermal management (*VADTM*) technique.

9.6. TV AND PV VARIATION AWARE MICRORING ASSIGNMENT (TPMA)

9.6.1. THERMAL VARIATION AWARE MR ASSIGNMENT (TMA)

As discussed in Section 3.1, TV shifts MR resonances, which can prevent MRs from reading or writing to their assigned carrier wavelengths. Fortunately, there is a linear dependency between temperature increase and resonance wavelength shift [162], which we exploit in our TV-aware

microring assignment (*TMA*) mechanism that dynamically assigns each MR to the nearest available carrier wavelength.

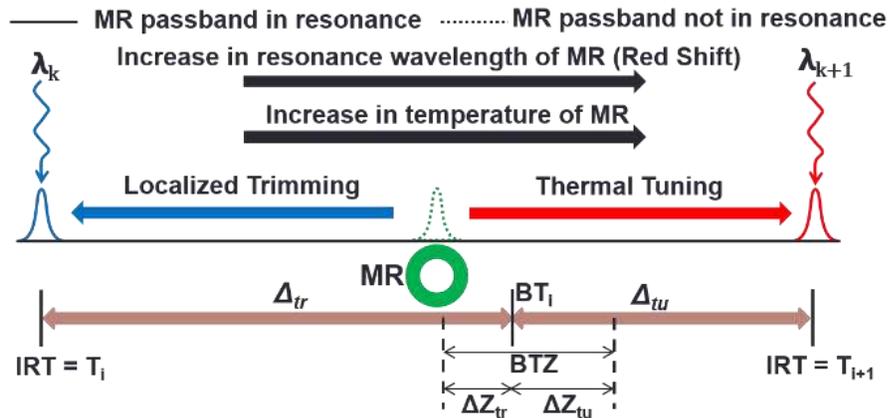


Figure 70 Red shift of MR with increase in temperature from IRTs T_i to T_{i+1} with trimming and tuning range of temperatures between these IRTs.

Figure 70 shows how at temperatures T_i and T_{i+1} ($T_{i+1} > T_i$), an MR resonance is in exact alignment with the available wavelengths λ_k and λ_{k+1} , respectively. These temperatures are called ideal resonant temperatures (IRTs). When the MR temperature is in between IRTs T_i and T_{i+1} , as shown in Fig.8, the MR needs to be either *trimmed* to resonate to λ_k (which is the resonance wavelength of an MR at temperature T_i) or thermally *tuned* to resonate to λ_{k+1} (which is the resonance wavelength of an MR at temperature T_{i+1}). To adaptively choose the least power consuming method from trimming and thermal tuning, we divide the temperature range between IRTs T_i and T_{i+1} into two parts: trimming temperature range (Δ_{tr}) and tuning temperature range (Δ_{tu}). For an MR at temperature T , if $(T_i + \Delta_{tr}) > T > T_i$ we perform trimming as it takes the least power, else if $(T_i + \Delta_{tr}) < T < T_{i+1}$ we perform tuning as it takes the least power (see Figure 70). At the boundary of the trimming and tuning temperature ranges, where $T_{i+1} - \Delta_{tu} = T_i + \Delta_{tr}$, both trimming and tuning consume equal power, and hence, an MR can be either trimmed or tuned.

This temperature is called the boundary temperature (BT_i). It has been shown that for a small resonance wavelength shift ($<1\text{nm}$), thermal tuning power is higher compared to trimming power to mitigate the same amount of TV-induced shift [19]. Thus, our TMA approach considers a higher trimming temperature range compared to tuning temperature range ($\Delta_{tr} > \Delta_{tu}$), to minimize total trimming and tuning power.

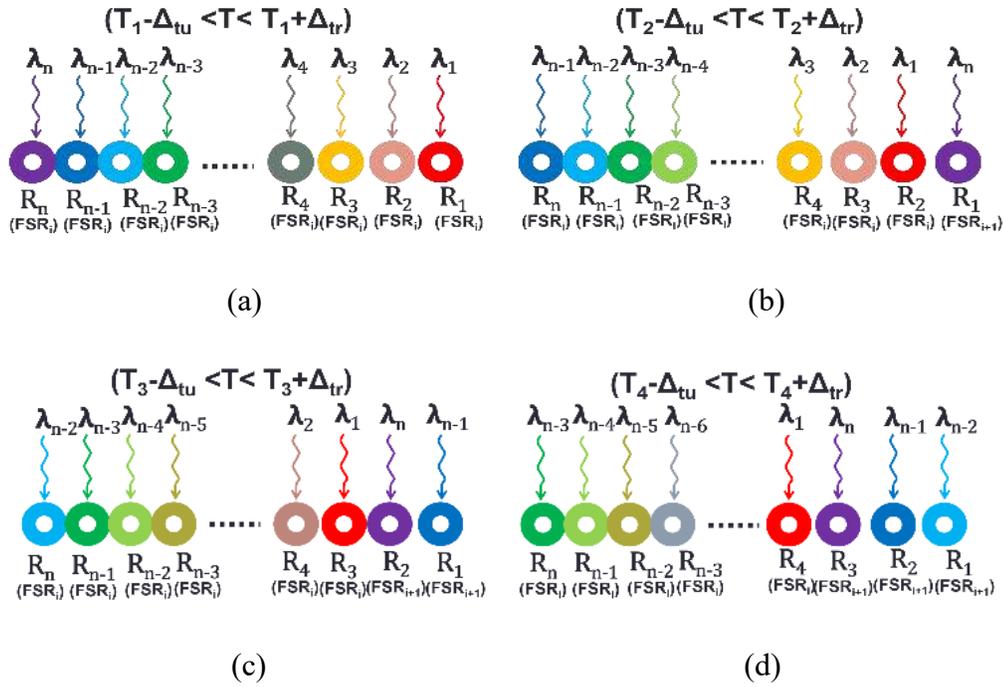


Figure 71 Thermal aware assignment of microrings (R_{1-n}) to wavelengths (λ_{1-n}) at four successive IRTs T_1, T_2, T_3 , and T_4 in TMA mechanism.

In TMA , MRs are dynamically shifted (trimmed or tuned) to an appropriate IRT for correct operation based on their current temperature. Figure 71(a)-(d) show four different MR wavelength assignment configurations at successive IRTs T_1, T_2, T_3 , and T_4 , where $T_4 > T_3 > T_2 > T_1$. If the MR group temperature T is such that $(T_1 - \Delta_{tu}) < T < (T_1 + \Delta_{tr})$ then the assignment in Fig. 9(a) is chosen, otherwise if $(T_2 - \Delta_{tu}) < T < (T_2 + \Delta_{tr})$, $(T_3 - \Delta_{tu}) < T < (T_3 + \Delta_{tr})$, or $(T_4 - \Delta_{tu}) < T < (T_4 + \Delta_{tr})$ then the assignment in Figure 71(b), Figure 71(c), or Figure 71(d) is chosen, respectively. One critical

observation in the assignment shown in Figure 71(a) is that MRs R_1 - R_n are in resonance with λ_{l-} λ_n within the same Free Spectral Range (FSR_i), whereas, in Figure 71(b) at IRT T_2 , MRs R_2 - R_n are in resonance with λ_{l-} λ_{n-1} , respectively in FSR_i and MR R_1 is in resonance with λ_n of the next FSR (i.e., FSR_{i+1}). In this assignment and the ones shown in Figure 71(c) and Figure 71(d), as explained in Section 4, there is a need to reposition bits in electrical domain using backend barrel-shifters or pipelined shift registers. The assignments shown in Figure 71(b), Figure 71(c), and Figure 71(d) require one, two, and three bit shifts, respectively, to retrieve the original data.

TMA represents a powerful reactive technique to adapt to on-die thermal variations with low overhead while ensuring reliable and high-bandwidth communication in MR based PNoCs. But there is scope for three further enhancements. First, *TMA* does not consider the impact of PV on MRs, thus there is a need to readapt *TMA* to address the impact of PV on MRs, which is discussed in subsection 9.6.2. Second, there is a need to proactively control the peak on-chip temperature to reduce the range of on-chip temperature swings, which ultimately limits the number of required bit shifts (this work caps the number bit shifts to three as shown Figure 71(d)) and reduces the latency to retrieve the original data. Third, at the BT temperature (Figure 70), maximum trimming or tuning power is required to realign the MR resonances to their nearest carrier wavelengths. Thus, avoiding BT temperatures at MRs can reduce trimming and tuning power overhead. As shown in Figure 70, we define a boundary temperature zone (BTZ) around each BT_i . This zone includes temperatures T such that $BT_i - \Delta Z_{tr} < T < BT_i + \Delta Z_{tu}$ where ΔZ_{tr} and ΔZ_{tu} are designer specified parameters. Cores with corresponding MR bank temperatures that are within BTZs are called boundary temperature cores (BTCs). As BTCs possess the highest trimming and tuning power overhead for their corresponding MR bank, a mechanism that reduces the number of BTCs can

save trimming and tuning power. Section 9.7 describes such a mechanism, which also controls the range of on-chip temperature swings within allowable limits.

9.6.2. READAPTING TMA FOR PROCESS VARIATIONS (PMA)

In this subsection, we readapt the *TMA* mechanism to address the impact of PV on MR resonances. When using the *TMA* mechanism, PV-induced red or blue shift ($\Delta\lambda_{PV}$) alters the resonance wavelength (λ_{BT_i}) of an MR at BT_i to λ_{BTR} or λ_{BTB} , respectively, as shown in Figure 72. This violates the actual definition of BT, which is the temperature from which either trimming to λ_k (which is the resonance wavelength of an MR at temperature T_i) or tuning to λ_{k+1} (which is the resonance wavelength of an MR at temperature T_{i+1}) dissipates equal power. For example, in case of a PV-induced blue shift, tuning λ_{BTB} to λ_{k+1} would consume more power than trimming it to λ_k , as λ_{BTB} is shifted towards λ_k from λ_{BT_i} . Therefore, in the *TMA* that is readapted for process variations (*PMA* mechanism), we propose to either increase or decrease the BTs in line with the PV-induced red or blue shifts in MR resonances, respectively.

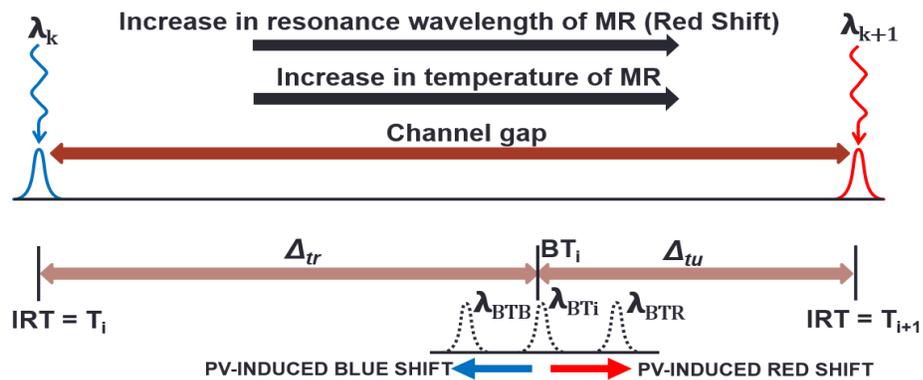


Figure 72 Impact of PV-induced red and blue shift on boundary temperature on *TMA*.

In our *PMA* mechanism, first, the PV-induced resonance shifts in MRs are gauged in situ at system initialization by using a dithering signal based control system [113]. The overhead of this in-situ PV detection technique is considered in our results section as dithering power. In our analysis, we model and estimate PV in MRs using the VARIUS tool [112], a description of which is already given in Section 9.3.3. Once PV-induced red or blue shifts of MRs are determined, we estimate the average resonance shift (in nm) across all MRs of each MR bank. We use each average shift value ($\Delta\lambda_{PV,ave}$) to determine the shift in BT (i.e., ΔBT_i) for all the MRs of the corresponding MR bank using Eq. (63), where TS is the MR thermal sensitivity obtained from Eq. (62) as $\Delta\lambda_r/\Delta T$.

$$\Delta BT_i = \frac{\Delta\lambda_{PV,ave}}{TS}, \quad (63)$$

Once the ΔBT_i values for all MR banks of the PNoC are obtained, we revise the BTs of each MR bank by either adding or subtracting the corresponding ΔBT_i value from the original BT. Similar to the *TMA* mechanism, we then build BTZs around these updated BTs. Note that we cannot shift the original BT beyond a particular temperature range (i.e. $\Delta BT_i > \Delta t_{tu}$ and $\Delta BT_i < -\Delta t_{tr}$), especially when the PV-induced resonance wavelength shifts are greater than one channel gap (CG). Unfortunately, for state-of-the-art fabrication processes, the maximum PV-induced wavelength shifts are around $\pm 3\text{nm}$ ($>$ one channel gap of 1.48nm). Shifting BT beyond a certain range to compensate for larger PV-induced shifts will also lead to higher tuning and trimming power dissipation.

Figure 73 shows an example of a larger PV-induced blue shift, which alters the resonance wavelength (λ_{BTi}) of an MR at BT to λ_{BTB} . One possible solution is to bring back the resonance wavelength to λ_{BTi} . But this is not always possible especially when the chip is operating at lower temperatures. Therefore, we propose to shift this λ_{BTB} to λ_{BTi-1} instead of λ_{BTi} , i.e., instead of

decreasing BT by a larger amount here we increase BT by a smaller amount. In order to facilitate this shifting, similar to *TMA*, we perform ring assignment along with extra bit shifts. At a channel spacing of 1.48nm, to compensate for peak PV-induced resonance shift of ± 3 nm, two extra bit shifts (forward and backward bit shifts to compensate positive and negative PV induced resonance shift) are needed.

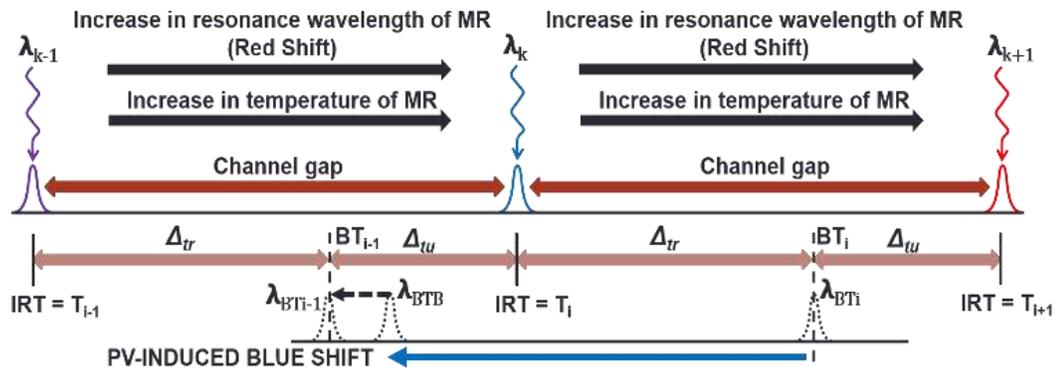


Figure 73 Boundary temperature adaptation for larger PV-induced blue shifts in *PMA*.

Overheads: Our proposed *TPMA* scheme is a combination of the two previously proposed techniques: *TMA* (Section 9.6.1) and *PMA* (this subsection). *TPMA* requires a maximum of five bit shifts, which include three for *TMA* and two for *PMA*. These additional bit shifts in *TPMA* incur latency overhead. This latency overhead is quantified in more detail in Section 9.8. Furthermore, with *TPMA* each MR bank requires a Read Only Memory (ROM) to store its corresponding three BT values, which are determined using PV profiling at design time, as discussed earlier. This ROM also stores beginning and ending temperatures of three BTZs in each MR bank. We have considered 16-bits to store each temperature value. As there is a need to store nine different temperature values (three BTs, three BTZ start temperatures, three BTZ end temperatures) for each MR bank, we need a ROM that can store 144-bits. Moreover, a 16-bit comparator circuit is needed

for each MR bank to determine the range of operation of MRs (i.e., trimming or tuning temperature range). This comparator is also used to determine whether an MR bank is in BTZ or not. Therefore, one input for this comparator comes from a thermal sensor (i.e., information on current temperature) and the other input is from the ROM. The area and power overhead of the ROM and comparator is quantified in detail in Section 9.8.

9.7. VARIATION AWARE ANTI WAVELENGTH-SHIFT DYNAMIC THERMAL MANAGEMENT (VADTM)

To proactively reduce thermal hotspots (which in turn will reduce instances of ‘irrecoverable shift’) and control on-die temperature (to reduce the number of BTCs), we propose a system-level variation aware anti wavelength-shift dynamic thermal management (*VADTM*) technique, described below.

9.7.1. OBJECTIVE

The primary goals with *VADTM* is to maintain the temperature of all of the cores on a die below a specified thermal threshold, i.e., for all cores $1 \leq i \leq N$, $T_i < T_t$ where T_i is the temperature of core i and T_t is threshold temperature. We utilize support vector based regression (SVR) to predict the future temperature of a core (for more details about this prediction model refer to Section 8.2.2.2). This predicted temperature is compared with a thermal threshold to determine the potential for a thermal emergency. If such a potential exists, threads are migrated to available BTCs. These BTCs are determined based on the PV profile of MRs and ring blocks that are used to send and receive data from these cores. Migration to a BTC has a twofold benefit. First by moving the thread away from a core that could suffer a thermal emergency, we avoid instances of irrecoverable

shift in the MR groups of that core. Second, by moving the thread to a BTC, the temperature of the BTC will increase resulting in that core no longer being a BTC (consequently the temperature of the core's MR groups will also increase, taking them outside of their BTZ and closer to IRTs, which will reduce trimming/tuning power). The parameters used to describe *VADTM* are shown in Table 22.

Table 22 List of VADTM parameters and their definitions

Symbol	Definition
IPC_i	Instructions per cycle of i^{th} core
CT_i	Current temperature of i^{th} core
TN_i	Average temperature of immediate neighboring cores of i^{th} core; if this core is on chip periphery and missing neighbors, then we consider virtual neighbor cores at ambient temperature in lieu of the missing cores
PT_i	Predicted temperature of i^{th} core
T_t	Thermal threshold
$BTCs$	Boundary temperature cores
$NBTCs$	Non-boundary temperature cores

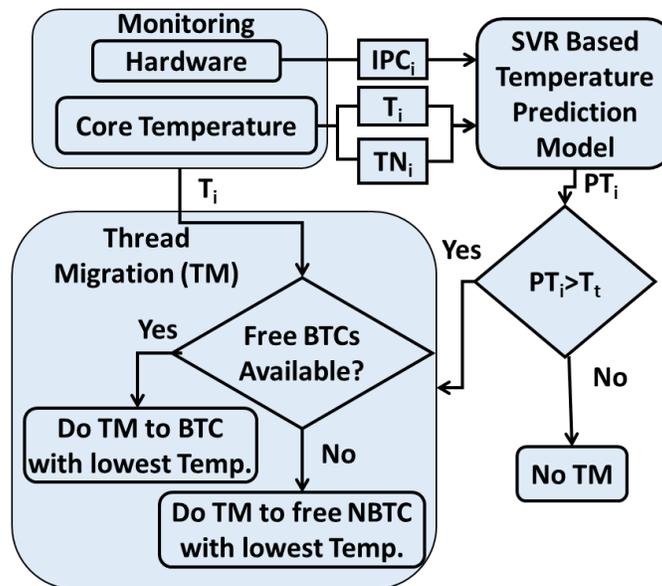


Figure 74 Overview of VADTM in LIBRA framework with support vector regression (SVR) based temperature prediction model.

9.7.2. THERMAL MANAGEMENT FRAMEWORK

Figure 74 illustrates the entire *VADTM* technique. For each core, we periodically monitor the IPC value from performance counters and temperature from on-chip thermal sensors. If a thermal emergency is predicted for a core by the SVR predictor, then *VADTM* initiates a thread migration procedure, otherwise no action is taken.

Algorithm 4 *VADTM* thread migration algorithm

Inputs: Current core temperature (CT_i), average neighboring core temperature (TN_i), current core IPC (IPC_i)

```
1: for each core i do // Loop that predicts future temperature
2:    $PT_i = SVR\_predict\_future\_temperature(CT_i, TN_i, IPC_i)$ 
3: end for
4: for each core i do // Loop that checks for free BTCs and NBTCs
5:   if  $CT_i$  in BTZ and  $IPC_i == 0$  then
6:     List_BTC = Push i //add core to BTC list
7:   else if  $IPC_i == 0$  then
8:     List_NBTC = Push i //add core to NBTC list
9:   end if
10: end for
11: for each core i do // Loop that performs thread migration
12:   if  $PT_i \geq T_t$  then
13:     if List_BTC  $\neq \{\}$  then
14:       Migrated_core = Find_min_temperature_core(List_BTC)
15:       Do_thread_migration( core_i  $\rightarrow$  Migrated_core)
16:       List_BTC = Pop i
17:     else if List_NBTC  $\neq \{\}$  then
18:       Migrated_core = Find_min_temperature_core(List_NBTC)
19:       Do_thread_migration( core_i  $\rightarrow$  Migrated_core)
20:       List_NBTC = Pop i
21:     end if
22:   end if
23: end for
```

Output: Thread migration to BTC or NBTC cores

Algorithm 4 shows the pseudo-code for the *VADTM* thread migration procedure. First, the future temperature (PT_i) of the i^{th} core is predicted using the SVR based predictor with inputs: core temperature (CT_i), core IPC (IPC_i), and temperature of neighboring cores (TN_i) in steps 1-3. The list of available BTCs (i.e., those that are not currently executing any thread) and available NBTCs is obtained in steps 4-10. In steps 11-12, a loop iterates over all cores and checks for possible thread

migration conditions (i.e., thermal emergency cases where current core predicted temperature (PT_i) is greater than thermal threshold (T_i)). If a thread migration is required, then in steps **13-21**, we check for free BTCs, and if they are available then we migrate the thread from the current core to the BTC with lowest temperature, else we migrate the thread to a free NBTC with lowest temperature. This *VADTM* thread migration procedure is invoked at every epoch (1ms).

9.8. EXPERIMENTAL RESULTS

9.8.1. EXPERIMENT SETUP

We target a 64-core manycore system to evaluate our *LIBRA* (*TPMA+VADTM*) framework. Each core has a Nehalem [x86](#) [141] micro-architecture with 32KB L1 instruction and data caches and a 256KB L2 cache, at 32nm and running at 5GHz. We evaluate *LIBRA* on two well-known PNoC architectures: Corona [11] and Flexishare [13]. Corona uses a 64×64 multiple write single read (MWSR) crossbar with token slot arbitration. Flexishare uses 32 multiple write multiple read (MWMR) waveguide groups with a 2-pass token stream arbitration. Each MWSR waveguide in Corona and MWMR waveguide in Flexishare is capable of transferring 512 bits of data from a source node to a destination node.

We modeled and simulated these architectures with the *LIBRA* framework for multi-threaded applications from the SPLASH-2 [131] and PARSEC [43] benchmark suites as explained in Section 8.2.2.2. Simulations were performed with a “warm-up” period of 100-million instructions and execution period of one billion cycles. Power and instruction traces for the benchmark applications were generated using the Sniper 6.0 [141] simulator and McPAT [142]. We used the 3D-ICE tool [130] for thermal analysis. The ambient temperature was set to 303K and the thermal threshold (T_i) was set to 353K.

We model and consider area, power, and performance overheads for our framework in our analysis. *LIBRA* with both Corona and Flexishare PNoCs has an electrical area overhead of 0.34 mm² and a power overhead of 57 mW using gate-level analysis and the CACTI 6.5 [114] tool for memory and comparators. The MR trimming power is set to 130μW/nm [18] for current injection (blue shift) and tuning power is set to 240μW/nm [19] for heating (red shift). To compute laser power, we considered detector responsivity as 0.8 A/W [74], MR through loss as 0.02 dB, waveguide propagation loss as 0.274 dB/cm, waveguide bending loss as 0.005 dB/90⁰, and waveguide coupler/splitter loss as 0.5 dB [74]. We calculated photonic loss in components using these values, which sets the photonic laser power budget and correspondingly the electrical laser power. For energy consumption of photonic devices, we adapt parameters from [74], with 0.42pJ/bit for every modulation and detection event, and 0.18pJ/bit for the driver circuits of MR modulators and photodetectors.

We also considered thread migration overhead in our simulations that ranged from 500-1000 cycles to account for startup latency (extra cache misses, branch mispredictions) in the migrated core. Further, our simulations considered PNoC latency to transfer data from architectural registers from the source core to the migrated core. This latency depends on locations of the cores and traffic conditions. As presented in Section 9.6, to minimize trimming and tuning power consumption, for a fixed channel gap of 1.48nm trimming temperature range (ΔZ_{tr}) and tuning temperature range (ΔZ_{tu}) for *TPMA* are calculated as 8.73K and 4.72K respectively. To minimize trimming and tuning power consumption further with lower performance overhead, there is a need to optimize ΔZ_{tr} and ΔZ_{tu} for *TPMA*. Therefore, we performed a sensitivity analysis to determine ΔZ_{tr} and ΔZ_{tu} values, as discussed in the next subsection.

9.8.2. SENSITIVITY ANALYSIS

Our first set of experiments involves a sensitivity analysis to explore the impact of the ΔZ_{tr} and ΔZ_{tu} parameters on *LIBRA*. We analyzed trimming and tuning power dissipation and execution time of the Flexishare PNoC with different values of these parameters. To be consistent with Δ_{tr} and Δ_{tu} , we consider the ratio of ΔZ_{tr} and ΔZ_{tu} to be equal to the ratio of Δ_{tr} and Δ_{tu} . For a fixed channel gap (i.e., 1.48nm), as presented above, the ratio of Δ_{tr} and Δ_{tu} is constant. Therefore, we determine the optimal ΔZ_{tu} with a sensitivity analysis and then we use that value to determine ΔZ_{tr} .

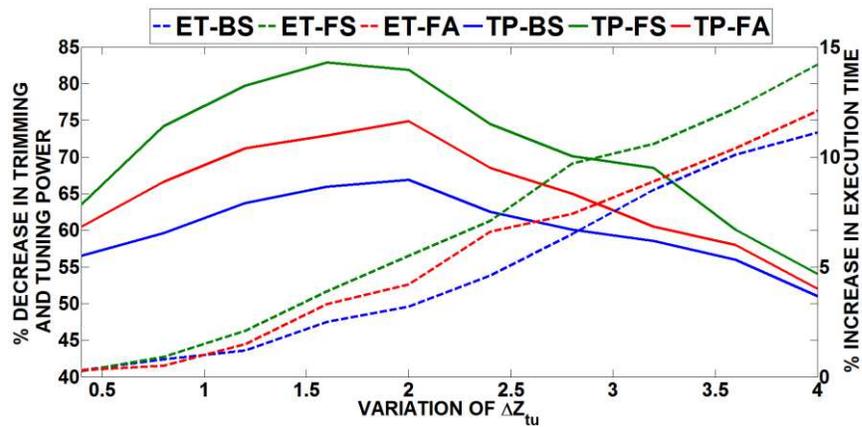
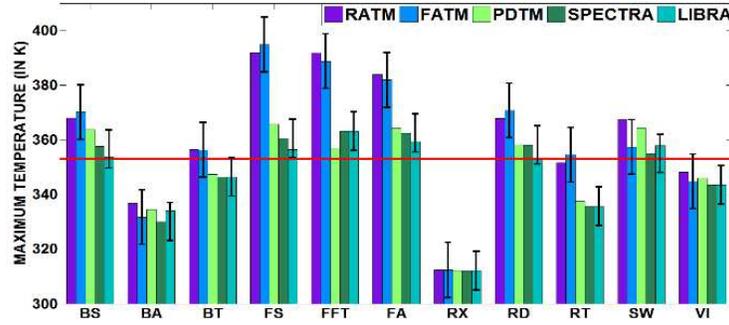


Figure 75 Percentage of decrease in trimming/tuning power (TP) and percentage of increase in execution time (ET) comparison across different ΔZ_{tu} values for *LIBRA* framework implemented on Flexishare PNoC in a 64-core CMP executing blackscholes (*BS*), Facesim (*FS*), and Fluidanimate (*FA*). Presented results are averaged across 100 PV maps. All percentage increments/decrements are calculated w.r.t baseline Flexishare PNoC employing frequency align scheduling policy (FATM).

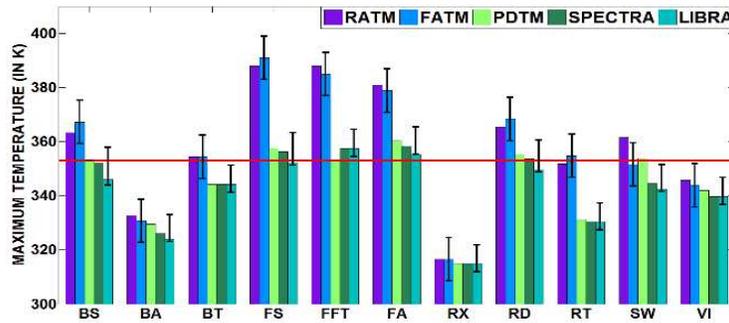
We considered 48-threaded *FS*, *FA*, and *BS* benchmark applications for our sensitivity analysis. Figure 75 shows the decrease in trimming and tuning power (TP) and increase in application execution time (ET) for the *LIBRA* framework while executing three benchmark applications on the Flexishare PNoC, with ΔZ_{tu} varying from 0.4K to 4K. We computed the decrease in TP and increase in ET with respect to baseline Flexishare PNoC architecture employing the FATM thread scheduling policy [145]. In this analysis, we presented results that are averaged

across 100 PV maps. The three benchmarks were chosen as they resulted in high (*FS*), medium (*FA*), and low (*BS*) peak temperatures, which allowed us to explore the impact of thread migration overheads on ΔZ_{tu} . At a particular ΔZ_{tu} , this figure shows higher TP savings for high peak temperature workloads (i.e., *FS*) compared to low peak temperature workloads (i.e., *BS*), as *LIBRA* effectively controls peak temperature and thereby reducing overall TP. Also the percentage of increase in application execution time is higher for high peak temperature workloads (i.e., *FS*) compared to low peak temperature workloads (i.e., *BS*), as *LIBRA* incurs more number of thread migrations for these workloads, which ultimately increases ET.

A careful observation of Figure 75 shows that for all the benchmark applications, *LIBRA*'s TP decreases with initial increase in ΔZ_{tu} and increases with further increase in ΔZ_{tu} . The main reason for this behavior is that at smaller values of ΔZ_{tu} *LIBRA* benefits by increasing temperature of BTCs, which ultimately reduces the number of MR groups within BTZs. Furthermore, larger values of ΔZ_{tu} increase BTZ size and the number of BTCs within it, so there is more chance that threads are migrated to cores whose temperatures are away from their BTs, which reduces the percentage of decrease in trimming and tuning power (TP; see Figure 75). Moreover, with increase in ΔZ_{tu} the number of thread migrations increase as more number of BTCs are available, which ultimately increases total execution time of the application (TP; see Figure 75). Thus, we set ΔZ_{tu} to 2K, to achieve higher TP savings with lower ET overhead. Using ΔZ_{tu} , as explained above, we determined ΔZ_{tr} as 3.7K. We used these values of the ΔZ_{tu} and ΔZ_{tr} parameters for our *LIBRA* framework in the rest of our analysis.



(a)



(b)

Figure 76 Maximum temperature comparison for *LIBRA* with RATM [133], FATM [145], PDTM [139] and SPECTRA [33], for (a) 48 thread, and (b) 32 thread PARSEC and SPLASH-2 benchmarks executing on 64-core manycore system with Corona PNoC. Bars show mean values of maximum temperature across 100 PV maps; confidence intervals show variation in maximum temperature.

9.8.3. COMPARISON RESULTS

We compared the performance of our *LIBRA* framework with four prior works on manycore thermal management: a ring aware policy (RATM) [133], frequency align policy (FATM) [145], a predictive dynamic thermal management (PDTM) framework [139], and the SPECTRA framework from our prior work [33]. RATM distributes threads uniformly across cores that are closer to PNoC nodes first and then distributes the remaining threads in a regular pattern from outer cores to inner cores. FATM distributes threads across cores based on the process variation profile of ring blocks that are in the proximity of these cores. PDTM uses a recursive least square

based temperature predictor to determine if the predicted temperature of a core exceeds a thermal threshold, and if so then thread migration is performed from that core to the coolest core that is not executing any threads. SPECTRA performs ring assignment at the device-level and SVR prediction based proactive thread migration at the system-level for thermal reliability management in PNOCs.

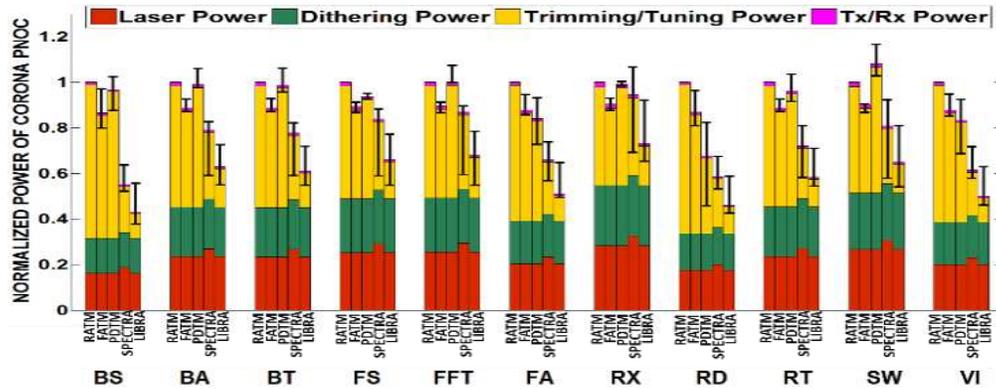
Figure 76(a)-(b) show the maximum temperature obtained with the five frameworks across eleven applications from the PARSEC and SPLASH-2 benchmarks suites with 48 and 32 thread counts executing on a 64-core system with the Corona PNoC [67] architecture. As *LIBRA* and *FATM* perform thread management based on the PV profile of MRs, only these frameworks have confidence intervals in Figure 76. From Figure 76(a) it can be observed that some applications (e.g., *FA*, *SW*) with 48 threads exceed the threshold (353K) for all frameworks, as there are insufficient number of free cores on the chip whose temperature is below the thermal threshold to migrate threads. However, with a more manageable number of threads, the situation improves. In Figure 76(b), for the case with 32 threads, our *LIBRA* framework avoids violating thermal thresholds for very small number of benchmark applications with 32 threads. On average, *LIBRA* has 14.6K and 17.5K lower maximum temperature compared to the *RATM* policy for 48 and 32 threads, respectively. In addition, on average *LIBRA* has 13.5K and 16.9K lower maximum temperature compared to the *FATM* policy for 48 and 32 threads, respectively. *LIBRA* migrates threads from hotter cores to cooler cores to control maximum temperature, whereas no thread migration is performed in both *RATM* and *FATM* when the on-chip thermal threshold temperature (i.e., 353K) is reached, as these mechanisms are simple thread allocation policies without control on peak temperature. For most of the benchmarks, maximum temperatures with *PDTM*, *SPECTRA*, and *LIBRA* are below the thermal threshold. However, on average *LIBRA* has 3.2K

and 3.5K lower maximum temperature compared to PDTM for 48 and 32 threads, respectively. This is because *LIBRA* employs a more accurate SVR based prediction approach which reduces the increase in peak temperature due to mispredictions, compared to the low accuracy of the least square regression mechanism in PDTM. Lastly, *LIBRA* has a 0.8K and 1.9K lower maximum temperature compared to SPECTRA for 48 and 32 threads, respectively. Even though both *LIBRA* and SPECTRA prefer to migrate threads to BTCs, the maximum temperatures with *LIBRA* are sometimes lower compared to SPECTRA, as *LIBRA* is able to perform thread migrations more often to lower temperature BTCs compared to SPECTRA.

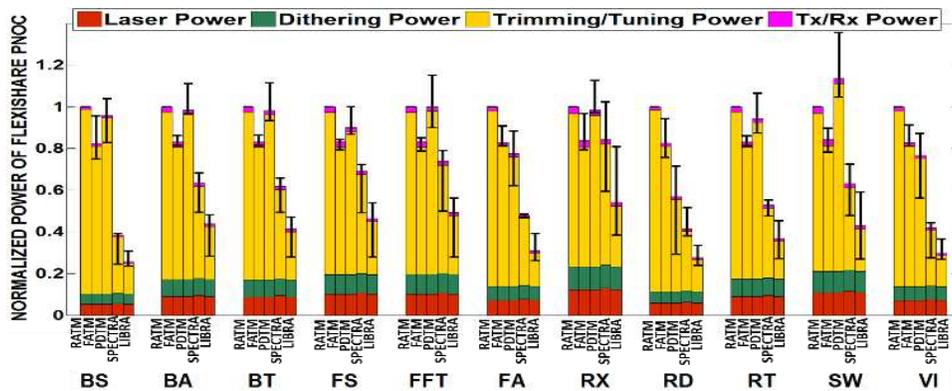
In the interest of brevity, we do not show maximum temperature results for the Flexishare PNoC architecture. We observed a similar trend in maximum temperature variations for Flexishare as we did for Corona (Figure 76).

Figure 77 shows the power dissipation comparison for the five frameworks across multiple 48-threaded applications for the Corona and Flexishare PNoC architectures, respectively. One of the main reasons why *LIBRA* has lower power dissipation than RATM, FATM, and PDTM is that it more aggressively reduces trimming and tuning power in both Corona and Flexishare PNoCs. From Figure 77(a), *LIBRA* has 74.5%, 67.4%, and 70.8% lower trimming and tuning power on average compared to RATM, FATM, and PDTM for Corona. Furthermore, from Figure 77(a), *LIBRA* also has 76.2%, 68.3%, and 72.5% lower trimming and tuning power on average compared to RATM, FATM, and PDTM for Flexishare. The *TPMA* technique in *LIBRA* intelligently conserves trimming and tuning power compared to RATM, FATM, and PDTM by performing process variation aware MR reassignment with increase in temperature, while our *VADTM* further improves trimming and tuning power savings with its intelligent thread migration to BTCs. Lastly, the *TPMA* mechanism in *LIBRA* adapts intelligently to the PV profiles of MRs, reducing it's

trimming and tuning power dissipation by 46.3% and 48.1%, compared to SPECTRA for the Corona and Flexishare architectures, respectively.



(a)

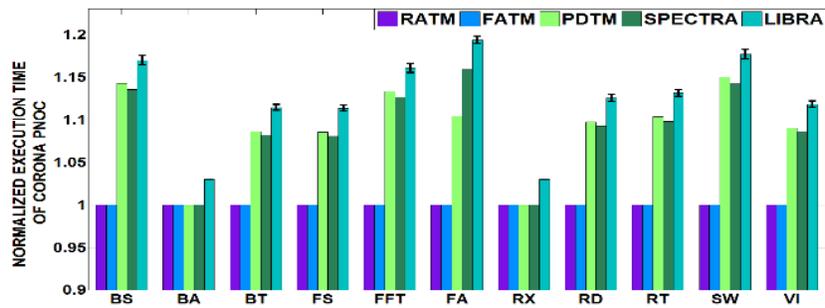


(b)

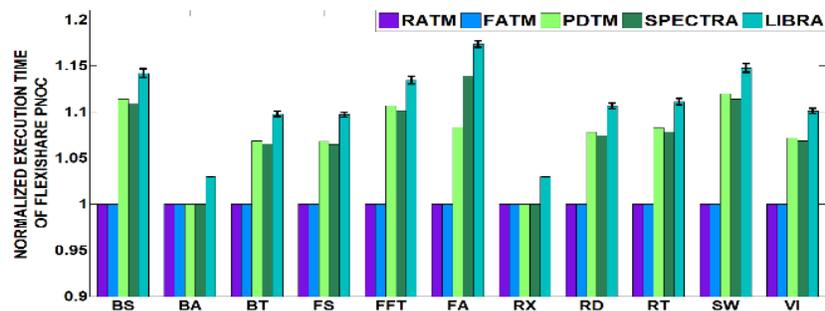
Figure 77 Normalized power dissipation (Laser Power, Dithering Power, Trimming/Tuning power, and Modulating and Detecting (Tx/Rx) Power) comparison for *LIBRA* with RATM [133], FATM [145], PDTM [139] and SPECTRA [33] for 48 threaded applications of PARSEC and SPLASH-2 suites executed on (a) Corona (b) Flexishare PNoC architectures for a 64-core manycore system. Results shown are normalized w.r.t RATM, therefore, RATM does not have confidence intervals. Bars show mean values of power dissipation across 100 PV maps; confidence intervals show variation in power dissipation.

Figure 77 also shows the laser power comparison of the five frameworks for the Corona and Flexishare architectures. It can be observed that Corona and Flexishare with *LIBRA* need similar laser power as Corona and Flexishare architectures with RATM, FATM, and PDTM. Furthermore,

LIBRA requires 12.9% and 6.4% lesser laser power compared to *SPECTRA* for Corona and Flexishare. The extra MRs used in *SPECTRA* to compensate for TV-induced resonance shifts contribute to the increase in laser power compared to *LIBRA* for both architectures. From these results it can also be observed that the laser power saving in Corona is higher than for the better performance optimized architecture of Flexishare.



(a)



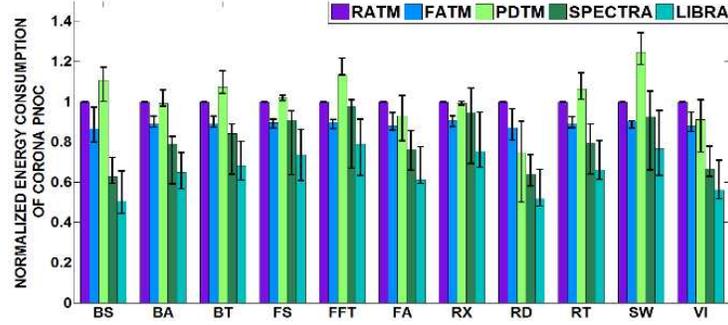
(b)

Figure 78 Normalized average execution time comparison of *LIBRA* with RATM [133], FATM [145], PDTM [139] and *SPECTRA* [33] for (a) Corona; and (b) Flexishare PNoCs for 48 threaded applications from PARSEC and SPLASH-2 suites executed on 64-core system. Results shown are normalized wrt RATM.

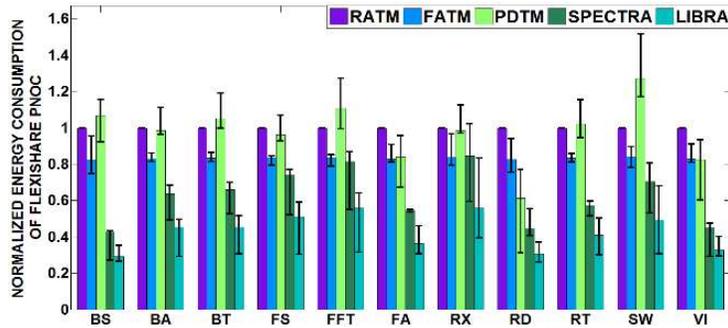
In summary, *LIBRA* saves considerable trimming/tuning power to ultimately achieve overall power reduction. From the power analysis in Figure 77(a), *LIBRA* with Corona has 40.8%, 34.1%, 37.2%, and 21.4% lower total power dissipation compared to Corona with RATM, FATM, PDTM, and *SPECTRA*, respectively. Further from Figure 77(b) it can be seen that Flexishare with *LIBRA*

has 61.3%, 52.9%, 57.4%, and 32.8% lower power dissipation compared to Flexishare with RATM, FATM, PDTM, and SPECTRA, respectively.

Figure 78 shows the average execution time comparison between the five frameworks across the 11 48-threaded applications from PARSEC and SPLASH-2 suites, for the Corona and Flexishare PNoC architectures, respectively. As only *LIBRA* performs thread migration based on the PV profile of MRs, therefore, this framework has confidence intervals on execution time shown in Figure 78. From Figure 78(a) it can be seen that Corona with *LIBRA* has 12.4% higher execution time compared to Corona with RATM and FATM. Corona with *LIBRA* needs extra execution time to migrate threads between cores and to reorder bits using shift registers whereas the RATM and FATM policies simply schedule threads without any thread migration and bit reorder, and thus do not possess such overheads. Further, Corona with *LIBRA* has 3.2% higher execution time compared to PDTM. Despite *LIBRA* using a faster SVR based temperature predictor compared to a more complex recursive least square based regression predictor in PDTM, the higher number of thread migrations (to adapt to PV profiles of MRs) and bit reordering operations in *LIBRA* contribute to an increase in execution time. Similarly, from Figure 78(b) Flexishare architecture with *LIBRA* has 10.6% higher execution time compared to Flexishare with RATM and FATM. In addition, *LIBRA* also has 2.8% higher execution time compared to Flexishare with PDTM. The figures also indicate that the execution time overheads for *LIBRA* are lower when utilizing the faster Flexishare architecture compared to the slower Corona architecture. Moreover, the bit-shifting overhead in *LIBRA* increases its execution time by 4.2% and 3.2% compared to the SPECTRA framework with Corona and Flexishare PNoCs, respectively. From the execution time results, it can be summarized that the significant power benefits achieved with *LIBRA* come at some cost: an increase in execution time.



(a)



(b)

Figure 79 Normalized energy consumption comparison of *LIBRA* with RATM [133], FATM [145], PDTM [139] and SPECTRA [33] for (a) Corona; and (b) Flexishare PNoCs for 48 threaded applications from PARSEC and SPLASH-2 suites executed on a 64-core system. Results shown are normalized wrt RATM, therefore, RATM does not have confidence intervals. Bars show mean values of energy consumption across 100 PV maps; confidence intervals show variation in energy consumption.

Lastly, from the power dissipation and execution time results, we obtain energy consumption results for the five frameworks, as shown in Figure 79. On average, for Corona, energy consumption with *LIBRA* is 34.5%, 25%, 35.4%, and 18.7% lower compared to RATM, FATM, PDTM and SPECTRA, respectively. For the Flexishare architecture, *LIBRA* has 57.3%, 47.9%, 55.6%, and 31.1% lower energy consumption compared to RATM, FATM, PDTM, and SPECTRA respectively.

In summary, from the above results, it is apparent that our proposed PV-aware *LIBRA* framework outperforms previously proposed approaches for thermal management in manycore

systems with PNoCs by combining a novel reactive device-level technique (*TPMA*) that improves waveguide channel utilization with a novel system-level proactive thread migration technique (*VADTM*). The excellent power and energy savings compared to previous approaches strongly motivate the use of our thermal management framework in future manycore architectures.

9.9. CONCLUSIONS

In this chapter, we have presented the *LIBRA* framework that combines two novel dynamic thermal management mechanisms for the reduction of maximum on-chip temperature and conservation of trimming and tuning power of MRs in DWDM-based PNoC architectures. These techniques (*TPMA* at the device-level, *VADTM* at the system-level) constitute a hybrid reactive-proactive management framework that demonstrated interesting trade-offs between performance and power/energy across two different state-of-the-art crossbar-based PNoC architectures. Our experimental analysis on the well-known Corona and Flexishare PNoC architectures has shown that *LIBRA* can notably conserve total power by up to 61.3% (trimming and tuning power by up to 76.2%) and total energy by up to 57.3%.

10. ANALYZING VOLTAGE BIAS AND TEMPERATURE INDUCED AGING EFFECTS IN PHOTONIC INTERCONNECTS FOR MANYCORE COMPUTING

To enable MRs to modulate and detect DWDM photonic signals, carrier injection in MRs through their voltage biasing is essential. But long-term operation of MRs with constant or time-varying temperature and voltage biasing causes aging. Such voltage bias temperature induced (VBTI) aging in MRs leads to resonance wavelength drifts and Q-factor degradation, which increases signal loss and energy delay product in photonic NoCs (PNoCs) that utilize photonic interconnects. This chapter explores VBTI aging in MRs and demonstrates its impacts on PNoC architectures for the first time. Our system-level experimental results on two PNoC architectures indicate that VBTI aging increases signal loss in these architectures by up to 7.6dB and increases EDP by up to 26.8% over a span of 5 years.

10.1. INTRODUCTION

MRs can be either in-resonance or out-of-resonance with respect to the utilized DWDM wavelengths. In resonance mode, an MR couples/removes light of the resonant wavelength from the waveguide, and hence, modulates logic “0” (represented by the absence of light in the waveguide) on the resonant wavelength. In contrast, in the out-of-resonance-mode, an MR does not couple any light from the waveguide, and hence, modulates logic “1” (represented by the presence of light in the waveguide) on the resonant wavelength. Thus, a particular sequence of 1s and 0s can be modulated on a wavelength by switching the corresponding MR off and on resonance with the wavelength in the same sequence. MRs can employ either voltage biasing [18] or heating [19] to switch from resonance-mode to out-of-resonance-mode or vice versa. *However, voltage*

biasing is preferred over heating [19] to switch resonance-modes of MRs, as it is faster and dissipates lower power.

To facilitate switching of resonance-modes of an MR with voltage biasing, a PN junction is created in the silicon (Si) core of the MR surrounded by silicon-di-oxide (SiO₂) cladding. A positive/negative voltage bias is applied to this PN-junction to inject/remove free carriers into/from the MR's Si core. For high frequency operation and lower power consumption, an MR's PN-junction is typically operated under a negative voltage bias or reverse bias [24] (otherwise known as carrier depletion mode of an MR). The application of this voltage bias generates an electric field across the MR's Si (core) and SiO₂ (cladding) boundary. Similar to MOSFETs, this electric field generates voltage bias temperature induced (VBTI) traps at the Si-SiO₂ boundary of the MR over time (i.e., VBTI aging). Our analysis has shown that these VBTI aging induced traps alter carrier concentration in the Si core of MRs, which incur resonance wavelength drifts and increase optical scattering loss in MRs to decrease Q-factor of MRs.

In this chapter, for the first time, we study the VBTI aging in MRs and its impact on PNoC architectures. At the device-level, we carefully developed analytical models for trap generation with VBTI aging in MRs. We also devise analytical models that determine variations of MR resonance wavelength shifts and Q-factor with aging-induced traps. These models are further extended to examine the impact of different operating temperatures and bias voltages, as well as process variations. From those models, we follow a mathematical bottom-up approach to analyze the system-level impact of aging on different PNoC architectures. We present our aging analysis on well-known Corona [11] and Clos [60] PNoCs running real-world multi-threaded PARSEC [43] benchmarks.

10.2. RELATED WORK

Recent research on silicon photonics for manycore computing has focused on exploring a wide spectrum of network topologies and protocols to enable efficient PNoC architectures [11], [13]. PNoCs utilize several photonic devices such as MRs as modulators and detectors, waveguides, splitters, and trans-impedance amplifiers (TIAs). The reader is directed to [28], [40] for more discussion on these devices.

Fabrication-induced process variations (PV) impact the cross section, i.e., width and height, of photonic devices such as MRs and waveguides [22], [166]. In MRs, PV causes resonance wavelength drifts, which can be counteracted by using device-level techniques such as voltage biasing (aka localized trimming) and heating (aka thermal tuning). On the other hand, thermal variations (TV) also alter the resonance wavelength of MRs, because of variations in refractive index of the core of MRs due to thermo-optic effects. Similar to PV, resonance wavelength drifts due to PV are compensated by voltage biasing and heating [19]. A few prior works have explored the impact of PV and TV on photonic links at the system-level [23], [105], [164]. In [23], a methodology to salvage network-bandwidth loss due to PV-drifts is proposed, which reorders MRs and trims them to nearby wavelengths. In [23], a thermal tuning based approach is presented that adjusts chip temperature using dynamic voltage and frequency scaling (DVFS) to compensate for chip-wide PV-induced resonance shifts in MRs. In [164] a tunable laser source design is demonstrated, in which the signal power at the source is adapted to compensate for signal losses due to temperature and process variations across photonic interconnects. *All of these works ignore the harmful effects of PV and TV remedies on aging in MRs.*

Aging has become an important reliability concern for ultra-scaled semiconductor devices with significant implications for both analog and digital circuit design. The most important aging

mechanisms in CMOS devices include bias temperature instability (BTI) aging and hot carrier injection (HCI) aging. BTI causes a threshold voltage increase in MOSFETs due to trap generation at the Si-SiO₂ interface [167]. Negative BTI (NBTI) is observed in pMOSFETs, and it usually dominates the positive BTI (PBTI) observed in nMOSFETs [167]. A few prior works have analyzed the impact of NBTI aging mechanisms on MOSFET devices at the device-level. Different hydrogen diffusion models are proposed in [168] to determine trap generation at the Si-SiO₂ interface of pMOSFETs. In [169] models for trap generation in the Si-SiO₂ interface of reduced cross-section MOSFETs (e.g., narrow-width planar MOSFET, triple gate MOSFET, surround-gate MOSFET) are presented. *However, none of these works considers the impact of aging on MRs and its implications on DWDM-based PNoCs.*

In view of the shortcomings of prior work, in this chapter we aim to analyze VBTI aging in MRs, quantify its dependence on temperature and bias voltage, and explore its impact at the PNoC architecture level.

10.3. TRIMMING (VOLTAGE BIAS) INDUCED MR AGING

10.3.1. OVERVIEW OF VOLTAGE BIAS INDUCED TRAP GENERATION IN MRS

MRs, waveguides, splitters, couplers, and TIAs are basic building blocks of PNoCs [9], [170]. MRs are essentially looped photonic waveguides with a small diameter (~a few μm), and these MRs serve as modulators to write data and detectors to read data. MRs when coupled to a waveguide in resonance-mode remove specific (resonant) wavelengths from the waveguide, whereas in the non-resonance-mode they let wavelengths simply pass through without removing them. MRs employ voltage biasing via carrier injection or removal to shift between resonance and non-resonance modes. To enable carrier injection into and removal from an MR, as shown in

Figure 80, a PN junction is created in an MR's Si core surrounded by SiO₂ cladding. To switch resonance modes at high frequency with low power dissipation using voltage biasing, an MR's PN junction needs to be reverse biased [24], which is accomplished by applying higher voltage on the n side of the PN junction (Figure 80).

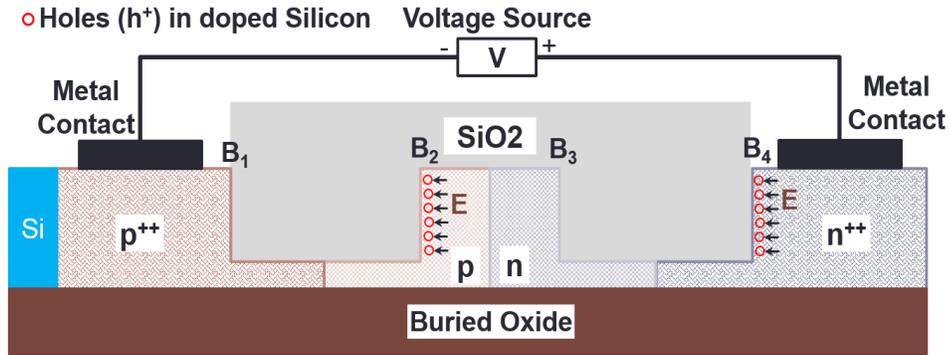


Figure 80 Cross-section of a tunable MR with PN junction in its core to facilitate carrier injection into and removal from core with voltage biasing.

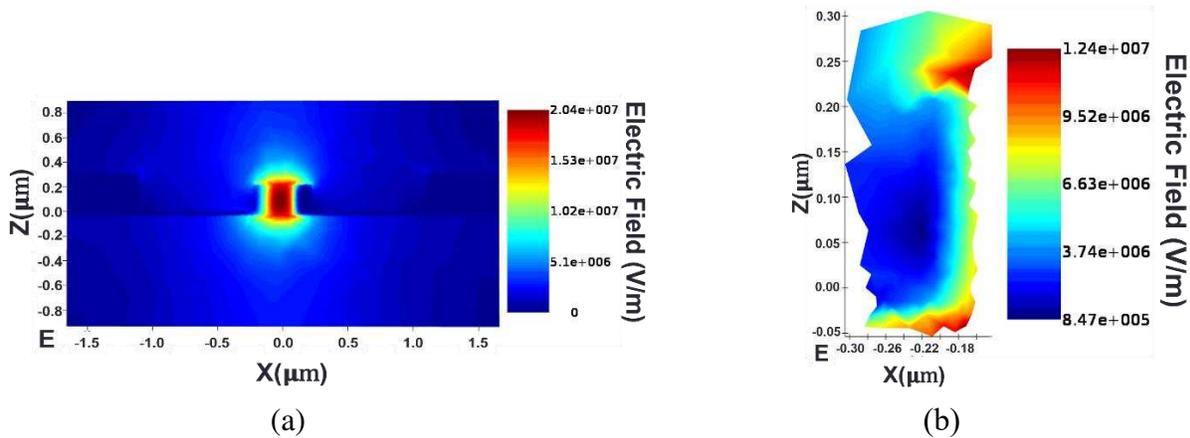


Figure 81 Distribution of electric field (E) across (a) MR waveguide; (b) Si-SiO₂ boundary B2 when -4V bias voltage is applied across PN junction.

When a negative voltage is applied across the PN junction of an MR, an electric field 'E' is generated from right to left across the Si-SiO₂ boundaries B₁, B₂, B₃, and B₄ (Figure 80). We used

the Lumerical Solutions DEVICE [122] tool to construct and model the PN junction of an MR. For our preliminary analysis, we consider an MR waveguide similar to the one reported in [110] with a radius of $2\mu\text{m}$, fabricated using standard Si-SiO₂ material with a core cross-section of $450\text{nm} \times 250\text{nm}$. We then simulated the MR, using the charge transport solver in the DEVICE tool with a solver geometry of 2D y-normal, and then obtained the distribution of electric field as shown in Figure 81(a) across the MR waveguide with a bias voltage of -4V. The results from the DEVICE tool in Figure 81(a) demonstrate the presence of electric field E across all the Si-SiO₂ boundaries (i.e., B₁, B₂, B₃, and B₄). This electric field present across the Si-SiO₂ boundaries B₂ (Figure 81(b)) and B₄ attracts holes towards them (Figure 80) and generates traps across these boundaries similar to pMOSFETs [167]. However, only the traps on the B₂ boundary change the electro-optic dynamics of the MR core as it is a boundary of the MR core. Thus in this chapter we focus on analyzing trap generation on the B₂ boundary.

10.3.2. TRAP GENERATION ANALYTICAL MODEL FOR MRS

The trap generation model on the B₂ boundary of an MR is based on Si-SiO₂ boundary related hydrogen dynamics [171]. The trap generation takes place at the Si-SiO₂ boundary which is a rough surface where the highly ordered Si core and the amorphous SiO₂ cladding meet. At the junction of these dissimilar materials, some of the Si atoms from the core remain dangling without satisfied chemical bonds, thus forming boundary traps. The traps generated at the Si-SiO₂ boundary of an MR are similar to the traps generated at the Si-SiO₂ boundary of a MOSFET [171]. To improve MR performance, there is a need to reduce these boundary traps. So similar to MOSFETs, MRs are annealed in ambient hydrogen during the manufacturing process. In the presence of an electric field and thermal variations across the Si-SiO₂ boundary, the Si-H bond

breaks and the hydrogen gas diffuses into the MR's SiO₂ cladding, thereby yielding passivated Si bonds (Si*) that act as traps. Furthermore, the direction of electric field (see Figure 80) across the MR's Si-SiO₂ boundary is similar to the direction of electric field across the MOSFET's Si-SiO₂ boundary. Therefore, at a particular temperature both MRs and MOSFETs have a similar trap generation behavior at their respective Si-SiO₂ boundaries.

Several prior works (e.g., [167]- [169]) use reaction-diffusion (RD) models to characterize boundary trap generation at the MOSFET Si-SiO₂ boundary. As boundary traps in MR's are similar to boundary traps in MOSFETs, we use the same RD model to model the boundary trap generation at the MR's Si-SiO₂ boundary. This trap generation mechanism is represented as a chemical reaction in Eq. (64), where holes (h⁺) in the MR's Si core weaken a Si-H bond and hydrogen (H) is detached [168] in the presence of electric field and thermal variations:



The generated Si dangling bond (Si*) acts as a donor-like boundary trap. The H ion released from the bond can diffuse away from the Si-SiO₂ boundary or anneal an existing trap. The boundary trap density (N_{BT}), increases with the net rate of the reaction given in Eq. (65):

$$\frac{dN_{BT}}{dt} = k_F[N_0 - N_{BT}] - k_R N_{BT} N_H \quad (65)$$

where k_F, k_R, N₀, and N_H are bond-breaking rate, bond-annealing rate, Si-H bond density available before stress, and hydrogen density at the MR's Si-SiO₂ boundary, respectively. From Eq. (65) it can be observed that the boundary trap generation rate increases with decrease in H ion density (N_H) at the Si-SiO₂ boundary. The diffusion of H ions away from the traps removes hydrogen from the boundary, so the boundary trap generation rate becomes limited to the diffusion rate of hydrogen. The diffusion rate of hydrogen obeys Eq. (66) [169]:

$$\frac{dN_H}{dt} = D_H \frac{d^2 N_H}{dy^2} \quad (66)$$

where D_H is the diffusion constant of hydrogen, dt is the change in time, and dy is the change in diffusion distance. During the diffusion-dominated regime, the dN_{BT}/dt term is negligible compared to the other two terms in Eq. (65) and N_{BT} is significantly smaller than N_0 [169], therefore Eq. (65) can be simplified as:

$$N_{BT}N_H = \frac{k_F N_0}{k_R} \quad (67)$$

Further, the dependence of the rate of boundary trap generation on the electric field across the boundary is included in the k_F term and the temperature dependence of trap generation is incorporated via the activation energies of k_F , k_R and D_H (see Sections 10.4, 10.5).

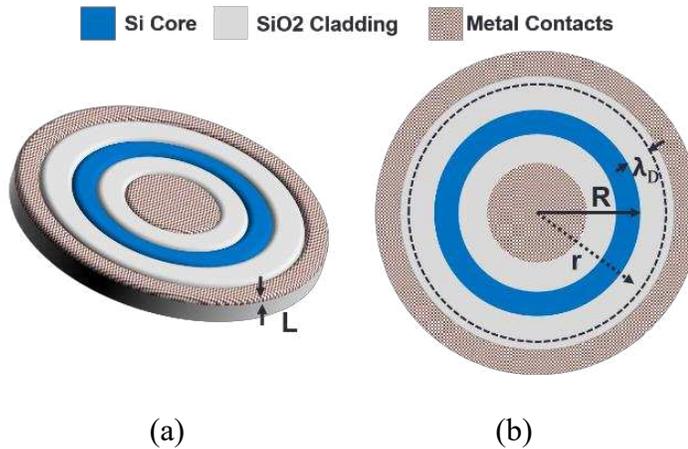


Figure 82 (a) Microring resonator 3D-view with Si-core, SiO₂-cladding, and metal contacts for voltage biasing; (b) top view of MR which shows hydrogen diffusion length (λ_D) across its cladding.

From the RD model presented above, the number of traps generated at the Si-SiO₂ boundary is equal to the number of hydrogen ions diffused away from the boundary. But this hydrogen diffusion depends on the geometry of the boundary. The effect of the geometry of hydrogen diffusion on the trap generation rate can be analyzed with the concept of the diffusion length λ_D , which is the distance travelled by hydrogen ion into SiO₂. As outer boundary (i.e. B₂) of an MR is

similar to the surround-gate cylindrical MOSFET [168], the λ_D of an MR is similar to this MOSFET. Therefore, based on estimations from prior works [168], [169] using Eq. (64) this diffusion length λ_D is estimated to be $(D_H^*t)^{0.5}$. For the MR with outer boundary (i.e. B₂) radius R and height or thickness L depicted in Figure 82, the hydrogen diffusion is confined within the distance $R < r < R + \lambda_D$, as shown in Figure 82(b). To determine the total hydrogen ions available within $R < r < R + \lambda_D$, there is a need to integrate all the hydrogen ions between R and $R + \lambda_D$. Thus the hydrogen profile is expressed in cylindrical coordinates and the integral becomes:

$$N_{BT}(t) = \frac{1}{2\pi RL} \int_R^{R+\lambda_D} N_H \left(1 - \frac{r-R}{\sqrt{D_H t}}\right) 2\pi r L dr \quad (68)$$

Solving Eq. (68) and substituting N_H from Eq. (67), the interface-trap density is calculated from the geometry-dependent R–D relation as:

$$N_{BT}(t) = \sqrt{\frac{k_F N_0}{k_R}} \left(\lambda_D \left(1 + \frac{R}{\lambda_D}\right) (2R + \lambda_D) - \left(\frac{R^2 + R(R + \lambda_D) + (R + \lambda_D)^2}{3}\right) \right)^{0.5} \quad (69)$$

From the above model it is clear that trap generation on an MR's Si-SiO₂ boundary not only depends on the operational time but also on the geometry of the boundary. *These traps are the main cause of aging in MRs. In the next subsection, we analyze how such boundary trap-induced aging impacts MR optical properties.*

10.3.3. AGING IMPACT ON MR RESONANCE WAVELENGTH AND Q-FACTOR

As discussed in the previous subsection, each trap generated on the core-cladding boundary of an MR consumes a hole from the P side of the MR core (Eq. (64)). Therefore, number of holes consumed in the silicon core is equal to number of boundary traps generated, which is otherwise $N_{BT} \approx -\Delta N_h$, where ΔN_h is the increase in free hole concentration and the negative sign represents decrease in free hole concentration. The removal of holes increases the refractive index of the core

(n_{Si}) of a circular MR waveguide, which induces a red shift in an MRs' resonance. The increase in the MR's core refractive index also increases refractive index contrast between the core and cladding ($n_{Si} - n_{SiO_2}$), which in turn increases the scattering related optical loss in the MR waveguide [172]. The increase in optical loss causes a decrease in MR Q-factor, which increases MR insertion loss. We quantify and model these phenomena in the rest of this section.

The change in hole concentration in an MR's core due to an MR aging induces refractive index change of Δn_{Si} at around 1550nm wavelength, which can be quantified as follows [111]:

$$\Delta n_{Si} = -8.8 \times 10^{-22} \Delta N_e - 8.5 \times 10^{-18} (\Delta N_h)^{0.8}, \quad (70)$$

where, ΔN_e and ΔN_h are the increase in free electron concentration and free hole concentration, respectively. Then, the increase in refractive index (positive n_{Si} as a result of aging-induced negative ΔN_h) incurs resonance wavelength red shift ($\Delta \lambda_{RWRS}$) as per the following equation [111]:

$$\Delta \lambda_{RWRS} = \frac{\Delta n_{Si} * \Gamma * \lambda_r}{n_g}, \quad (71)$$

where, λ_r is the initial resonance wavelength of the MR, n_g is the group refractive index (ratio of speed of light to group velocity of all wavelengths traversing the waveguide) of the MR, and Γ is the confinement factor describing the overlap of the optical mode with the MR waveguide's silicon core. The value of Γ and n_g for MR considered in our analysis are set to 0.7 and 4.2 respectively [110]. From [111], n_g accounts for refractive index dispersion and change in free carrier concentration (and hence, aging) does not significantly affect it.

An increase in the MR core's refractive index (Δn_{Si}) also increases its scattering loss coefficient. The scattering loss coefficient (that causes a fractional loss in signal amplitude) of an MR's circular waveguide is proportional to the size of the surface roughness σ , and is given by the following equation [119] [126]:

$$\alpha_{scatter} = \frac{4(\cos \theta)^3 k_0^2 n_1^2 \sigma^2}{\sin \theta} \cdot \left(\frac{k_0 \sqrt{n_1^2 (\sin \theta)^2 - n_2^2}}{L k_0 \sqrt{n_1^2 (\sin \theta)^2 - n_2^2} + 2} \right) \quad (72)$$

where, k_0 is the free-space wave number at 1550nm, $n_1 = n_{Si} = 3.5$ is MR core's refractive index, $n_2 = n_{SiO_2} = 1.5$ is MR cladding's refractive index, $L = 250\text{nm}$ is the MR thickness, and $\theta = 26.51$ is the propagation angle for the fundamental mode in the MR. $\alpha_{scatter}$ corresponding to an increase in the MR core's refractive index (Δn_{Si}) can be evaluated from Eq. (72) by putting $n_1 = n_{Si} + \Delta n_{Si}$ in it.

The Q-factor of an MR with resonance wavelength (λ_r) depends on this scattering loss coefficient. The relation between the Q-factor and $\Delta \alpha_{scatter}$, assuming critical coupling of MRs, is given by the following equation [110], where Q_A is the loaded Q-factor of the aged MR:

$$Q_A = Q + \Delta Q = \frac{\pi n g}{\lambda_r (\alpha + \Delta \alpha_{scatter})}, \quad (73)$$

where, ΔQ is the change in Q-factor and α is the original loss coefficient, which is the sum of three components: (i) intrinsic loss coefficient due to material loss and sidewall roughness induced scattering loss; (ii) bending loss coefficient, which is a result of the curvature in the MR; and (iii) the absorption effect factor that depends on the original free carrier concentration in the waveguide core. As explained above, aging increases the scattering loss coefficient (positive $\Delta \alpha_{scatter}$). As evident from Eq. (73), a positive value of $\Delta \alpha_{scatter}$ results in a decrease in Q-factor. This causes a broadening of the MR passband, which results in increased insertion loss.

For our VBTI aging analysis with MRs, we have considered initial original Q-factor of 9000 and loss coefficient α of 9.5cm^{-1} . As mentioned earlier, α is the sum of the scattering loss coefficient $\alpha_{scatter}$, bending loss coefficient α_b , and absorption loss coefficient α_a , the initial values of which, in this case (for $\alpha = 9.5\text{cm}^{-1}$), are 3.5cm^{-1} , 3cm^{-1} , and 3cm^{-1} respectively. Note that $\alpha_{scatter} = 3.5\text{cm}^{-1}$ corresponds to $\sigma = 5\text{nm}$ in Eq. (72).

10.4. TEMPERATURE INDUCED MR AGING

Aging in MRs is also dependent on the operating temperature (T) of the devices. As temperature alters activation energy for the Si–H bond breaking and bond annealing, it alters the bond-breaking rate (k_F) and bond-annealing rate (k_R) of the reaction shown in Eq. (64). We use the Arrhenius equation [173] to determine variation in activation energies with temperature. Eq. (74) and Eq. (75) present the temperature dependence of k_F and k_R respectively:

$$k_F = k_{F0} e^{\frac{-E_F}{K_B T}} \quad (74)$$

$$k_R = k_{R0} e^{\frac{-E_R}{K_B T}} \quad (75)$$

where, E_F and E_R are activation energies of forward dissociation and reverse annealing respectively, and K_B is the Boltzmann constant. The activation energy (E_D) of diffusion of hydrogen into the cladding of MRs also depends on temperature, which in turn alters the diffusion constant of hydrogen (D_H) as per the following equation:

$$D_H = D_0 e^{\frac{-E_D}{K_B T}} \quad (76)$$

Figure 83 shows the variation of resonance wavelength red shift ($\Delta\lambda_{RWRS}$) and Q_A with aging in MRs at different temperatures. We analyze $\Delta\lambda_{RWRS}$ and Q_A across different operating temperatures: 300K, 350K, and 400K. From the figure it can be observed that at a particular temperature, with the increase in MR aging (i.e., increase in usage time) $\Delta\lambda_{RWRS}$ increases and Q_A decreases. With MR aging, the traps on the Si-SiO₂ boundary increase, which is evident from Eq. (69). Furthermore, change in temperature also alters k_F , k_R , and D_H as per Eq. (74), (75), and (76), respectively. These rate constants ultimately change the number of traps generated at the Si-SiO₂ boundary as per Eq. (69). An increase in number of traps incurs an increase in refractive index of an MR (see Eq. (70)), which in turn increases the MR's $\Delta\lambda_{RWRS}$ (see Eq. (71)) and scattering loss

(α_{scatter}) (see Eq. (72)). Increase in α_{scatter} decreases an MR's Q_A as per Eq. (73). From the figure we can also observe a higher increase in $\Delta\lambda_{\text{RWRS}}$ and higher decrease in Q_A with an increase in MR's operating temperature. As the temperature increases, the activation energy (E_D) of diffusion of hydrogen (see Eq. (76)) in the cladding of an MR decreases, which increases the diffusion rate of hydrogen and further increases trap generation at the MR core-cladding boundary. This increase in number of traps ultimately leads to higher increase in RMRS and higher decrease in Q_A .

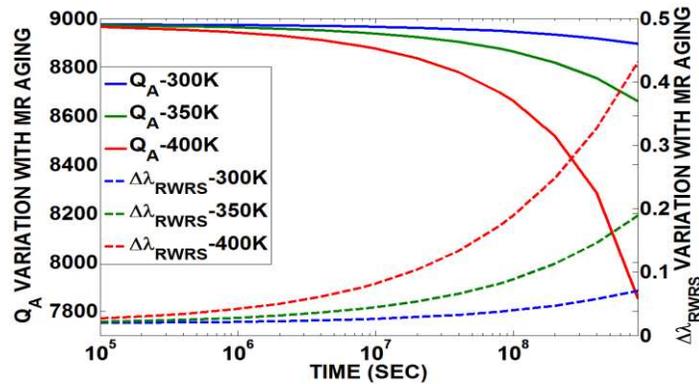


Figure 83 Variation of resonance wavelength red shift ($\Delta\lambda_{\text{RWRS}}$) and Q_A with operation time at three operating temperatures 300K, 350K, and 400K.

10.5. IMPACT OF PROCESS VARIATIONS ON MR AGING

Variations in an MR's width and thickness due to process variations (PV) cause a "shift" in the resonance wavelength of the MR. As discussed earlier, voltage biasing (aka localized trimming) is essential to deal with PV-induced resonance shifts in MRs. There are other techniques such as thermal tuning that used to compensate PV-induced resonance drifts. However, thermal tuning has higher power overhead (240 $\mu\text{W}/\text{nm}$) to compensate 1nm PV-induced drift compared to localized trimming (130 $\mu\text{W}/\text{nm}$) [19]. Therefore, voltage biasing or trimming is preferred to compensate PV-induced resonance drifts over thermal tuning. Voltage biasing incurs blue shift/red

shift in an MR's resonance wavelength via carrier injection/removal. To enable localized trimming in MRs to counteract PV-induced blue shifts, the negative bias voltage needs to be increased across the MR's reverse-biased PN junction. Unfortunately, this PV-induced increase in negative bias voltage results in an increase in the electric field across the MR core-cladding boundary and this electric field aggravates MR aging.

The forward dissociation constant (k_F) in Eq. (65) will depend on the electric field across core-cladding boundary (E_{OX}). Thus the equation for k_F shown in Eq. (74) is updated as per the following equation [167]:

$$k_F = B\sigma_0 E_{OX} e^{\frac{E_{OX}}{E_0}} e^{\frac{-E_F}{k_B T}} \quad (77)$$

where $\exp(E_{OX}/E_0)$ is the field dependent tunneling of holes into SiO_2 cladding, σ_0 is the capture cross-section of the Si-H bonds, and B determines field dependence of the Si-H bond dissociation.

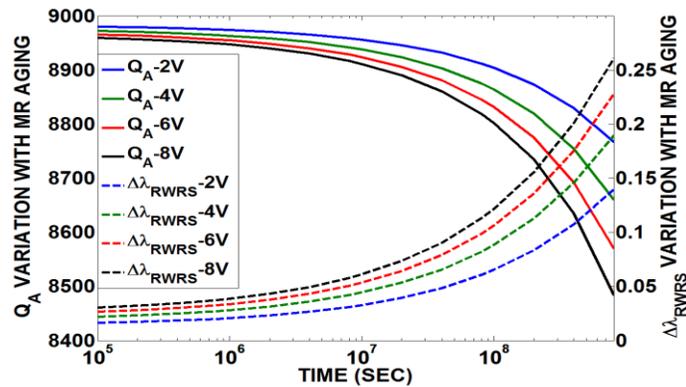


Figure 84 Variation of Q_A and resonance wavelength red shift ($\Delta\lambda_{RWRS}$) with operation time at four bias voltages -2V, -4V, -6V, and -8V.

Figure 84 illustrates the impact of variation in bias voltage on $\Delta\lambda_{RWRS}$ and Q_A of MR with aging (i.e., usage time). We analyze negative voltage biases of 2V, 4V, 6V, and 8V, and the MR is assumed to be operated at 350K temperature. As explained Section 10.3.1, the charge transport

solver in the DEVICE tool is used to determine electric field (E_{OX}) across the core-cladding boundary for each bias voltage across the PN junction of the MR. This tool uses MR device dimensions such as width, height and radius to determine E_{OX} at the boundary. From the figure it can be observed that with the increase in negative bias voltage, MRs incur higher $\Delta\lambda_{RWRS}$ increase (see Eq. (71)) and higher Q_A decrease (See Eq. (72)). As the negative bias voltage across the PN junction of the MR increases, the E_{OX} across the core-cladding boundary of the MR increases. This increase in E_{OX} increases k_F as per Eq. (77), which in turn increases trap generation across the core-cladding boundary as per Eq. (69). This increase in trap generation increases $\Delta\lambda_{RWRS}$ and Q_A of an MR, as also highlighted by the Eq. (70)-(73) presented in Section 10.3.3.

Table 23 Notations for photonic power loss and model parameters [28]

Notation	Parameter type	Parameter value (in dB)
L_P	Propagation loss	-0.274 per cm
L_B	Bending loss	-0.005 per 90°
L_{S12}	1X2 splitter power loss	-0.2
L_{S14}	1X4 splitter power loss	-0.2
L_{S17}	1X7 splitter power loss	-0.2
L	Photonic path length in cm	
B	Number of bends in photonic path	
λ_r	Resonance wavelength of MR	
R_{S12}	Splitting factor for 1X2 splitter	
R_{S14}	Splitting factor for 1X4 splitter	
R_{S17}	Splitting factor for 1X7 splitter	

10.6. IMPACT OF MR VBTI AGING ON PNOCS

10.6.1. MR AGING ANALYSIS FOR CORONA AND CLOS PNOCS

We characterize the impact of VBTI aging on two popular PNoC architectures: Corona [11] and Clos [60], both of which use DWDM-waveguides for data communication. We have considered Corona PNoC with token-slot arbitration [67] and an 8-ary 3-stage Clos PNoC [60] for our analysis. In DWDM-based waveguides, data transmission requires modulating light using a

group of MR modulators equal to the number of wavelengths supported by DWDM. Similarly, data detection at the receiver requires a group of detector MRs equal to the number of DWDM wavelengths. We present analytical equations to model the impact of aging on maximum signal power loss in each architecture. Before presenting relevant equations, we provide notations for the parameters used in the equations, in Table 23.

We first model the MR transmission spectrum at a device-level and then extend these models to the system-level to determine the impact of aging on signal losses for PNoC architectures. We model the MR transmission spectrum using a Lorentzian function [124]. In Eq. (78), this function is used to represent coupling factor Φ between wavelength λ_i and an MR with resonance wavelength λ_j . Further, using the same function, we determined loss factor γ in Eq. (79), which is the factor by which signal power of a wavelength λ_i is reduced when it passes through an MR whose resonance wavelength is λ_j . Through loss of a wavelength in a waveguide, when it passes through an MR, is defined as γ times the signal power of the wavelength before it passes through the MR. From Eq. (72) and (73), it can be inferred that an MR's loaded Q-factor (Q_A) decreases with aging in MRs. This in turn decreases Φ and increases γ as per Eq. (78) and (79), respectively. Furthermore, as per Eq. (78) and (79) increase in $\Delta\lambda_{RWRS}$ with aging (i.e., $\Delta\lambda_{RWRSAi}$) further decreases Φ and increases γ , respectively.

$$\Phi(\lambda_i, \Delta\lambda_{RWRS Ai}, \lambda_j, Q_A) = (1 + (\frac{2Q_A(\lambda_i + \Delta\lambda_{RWRS Ai} - \lambda_j)}{\lambda_j})^2)^{-1}, \quad (78)$$

$$\gamma(\lambda_i, \Delta\lambda_{RWRS Ai}, \lambda_j, Q_A) = (1 + (\frac{2Q_A(\lambda_i + \Delta\lambda_{RWRS Ai} - \lambda_j)}{\lambda_j})^{-2})^{-1}, \quad (79)$$

Corona PNoC: This PNoC is designed for a 256 core single-chip platform, where cores are grouped into 64 clusters, with 4 cores in each cluster. A photonic crossbar topology with 64 data channels is used for communication between clusters. Each channel consists of 4 multiple-write-

single-read (MWSR) waveguides with 64-wavelength DWDM in each waveguide. As modulation occurs on both positive and negative edges of the clock in Corona, 512 bits (cache-line size) can be modulated and inserted on 4 MWSR waveguides in a single cycle by a sender. A data channel starts at a cluster called ‘home-cluster’, traverses other clusters (where modulators can modulate light and detectors can detect this light), and finally ends at the home-cluster again, at a set of detectors (optical termination). A power waveguide supplies optical power from an off-chip laser to each of the 64 data channels at its home-cluster, through a series of 1X2 splitters. In each of the 64 home-clusters, optical power is distributed among 4 MWSR waveguides equally using a 1X4 splitter with splitting factor R_{S14} . As all 1X2 splitters are present before the last (64th) channel, this channel suffers the highest signal power loss. Thus, the worst-case signal loss exists in the detector group of the 64th cluster node, and this node is defined as the worst-case power loss node (N_{WCPL}) in the Corona PNoC. For this N_{WCPL} node, signal power ($P_{signal}(\lambda_j)$) on each detector with resonance wavelength λ_j is shown in Eq. (80). $K(\lambda_i)$ in Eq. (82) represents signal power loss of λ_i before the detector group of N_{WCPL} (see Table 1 for notations of different parameters). $\psi(\lambda_i, \lambda_j)$ in Eq. (81) represents signal power loss of λ_i before the detector with resonance wavelength λ_j in the detector group of N_{WCPL} .

$$P_{signal}(\lambda_j) = K(\lambda_i)\psi(\lambda_i, \lambda_j) \Phi(\lambda_j, \Delta\lambda_{RWRS Aj}, \lambda_j, Q_{A(63 \times 64) + j}) P_{in}(i), \quad (80)$$

$$\psi(\lambda_i, \lambda_j) = \prod_{k=1}^{(k-1) < j} \gamma(\lambda_i, \Delta\lambda_{RWRS Ai}, \lambda_k, Q_{A(63 \times 64) + k}), \quad (81)$$

$$K(\lambda_i) = (R_{S14})(L_{S14})(L_P)^L(L_B)^B \prod_{n=1}^{63} \prod_{j=1}^{64} \gamma(\lambda_i, \Delta\lambda_{RWRS Ai}, \lambda_j, Q_{A((n-1) \times 64) + j}) \quad (82)$$

Clos PNoC: An 8-ary 3-stage Clos topology is considered for a 256-core system, with 8 clusters (C1-C8) and 32 cores in each cluster. Within each cluster, a group of four cores are connected to a concentrator. Thus each cluster has 8 concentrators and the concentrators are connected electrically through a router for inter-concentrator communication. The Clos PNoC uses photonic signals for

inter-cluster communication. Unlike the MWSR waveguides used in the Corona crossbar, the Clos uses point-to-point photonic links for data communication. Each point-to-point photonic link uses either forward or backward propagating wavelengths depending on the physical location of the source and destination clusters. Each photonic link in the Clos PNoC use 128 DWDM, with 64 wavelengths for forward communication and the remaining 64 wavelengths for backward communication. Thus the Clos PNoC uses only 56 waveguides with 256 MRs on each waveguide. This PNoC uses 2 laser sources to enable forward and backward communication. To power the 56 waveguides, it is assumed that the PNoC employs a series of 1X2, 1X7, and 1X4 splitters. In our implementation of the Clos PNoC, the worst-case power loss occurs when C1 sends data to C8, as this involves the longest photonic path for data traversal. Thus the node C8 is the worst-case power loss node (N_{WCPL}) in the Clos PNoC. We use Eq. (80) to determine worst-case power loss in the Clos PNoC. But as the Clos network has lower number of waveguides and fewer number of MRs on each waveguide, this in turn changes the signal power losses. Thus we modify Eq.(82) for the Clos PNoC as:

$$K(\lambda_i) = (R_{S14})(L_{S14})(L_P)^L(L_B)^B \prod_{n=1}^3 \prod_{j=1}^{64} \gamma(\lambda_i, \Delta\lambda_{RWRSai}, \lambda_j, Q_{A((n-1)\times 64)+j}) \quad (83)$$

10.6.2. MODELING PV OF MR DEVICES IN CORONA AND CLOS PNOCS

We adapt the VARIUS tool [112] to model die-to-die (D2D) as well as within-die (WID) process variations in MRs for the Corona and Clos PNoCs. VARIUS uses a normal distribution to characterize on-chip D2D and WID process variations. The key parameters are mean (μ), variance (σ^2), and density (α) of a variable that follows the normal distribution. As wavelength variations are approximately linear to dimension variations of MRs, we assume they follow the same distribution. The mean (μ) of wavelength variation of an MR is its nominal resonance wavelength.

We consider a DWDM wavelength range in the C and L bands [104], with a starting wavelength of 1550nm and a channel spacing of 0.8nm. Hence, those wavelengths are the means for each MR modeled. The variance (σ^2) of wavelength variation is determined based on laboratory fabrication data [14] and our target die size. We consider a 256-core chip with die size 400 mm² at a 22nm process node. For this die size we consider a WID standard deviation (σ_{WID}) of 0.61nm [105] and D2D standard deviation (σ_{D2D}) of 1.01 nm [105]. We also consider a density (α) of 0.5 [105] for this die size. With these parameters, we use VARIUS to generate 100 PV maps, each containing over 1 million points indicating the PV-induced resonance shift of MRs. The total number of points picked from these maps equal the number of MRs in the Corona and Clos PNoCs.

10.7. EXPERIMENTS

10.7.1. EXPERIMENT SETUP

We evaluate the impact of VBTI aging on PNoCs on the Corona and Clos PNoC architectures. We modeled and performed simulation based analysis of the Corona and Clos PNoCs using a cycle-accurate NoC simulator, for a 256 core single-chip architecture at 22nm. As explained in Section 10.6.2, we generated 100 PV maps to evaluate MR aging impact on these PNoCs for different PV profiles. We used real-world traffic from applications in the PARSEC benchmark suite [43]. GEM5 full-system simulation [72] of parallelized PARSEC applications was used to generate traces that were fed into our cycle-accurate NoC simulator. We set a “warm-up” period of 100 million instructions and then captured traces for the subsequent 1 billion instructions. We performed geometric calculations for a 20mm×20mm chip size, to determine lengths of MWSR waveguides in the Corona PNoC and photonic links in the Clos PNoC. We

consider a 5 GHz clock frequency of operation for the cores. A 512-bit packet size is utilized for both Corona and Clos PNoCs.

The static and dynamic energy consumption of electrical routers and concentrators in the Corona and Clos PNoCs is based on results from the open source DSENT tool [75]. For energy consumption of photonic devices, we adapt model parameters from recent work [73], [74], with 0.42pJ/bit for every modulation and detection event and 0.18pJ/bit for the driver circuits of modulators and photodetectors. We used optical loss in photonic components (Table 23) to estimate the photonic laser power budget and correspondingly the electrical laser power [39].

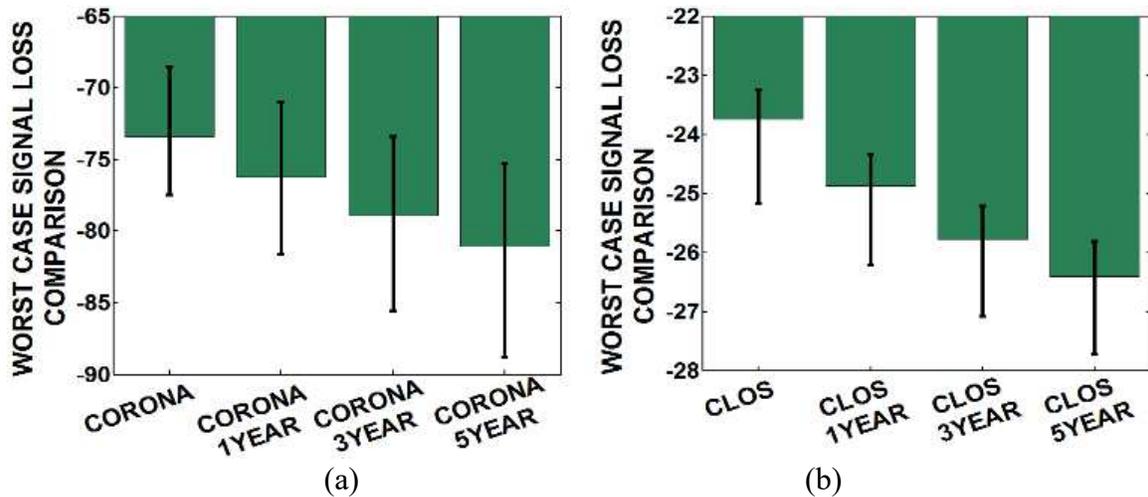
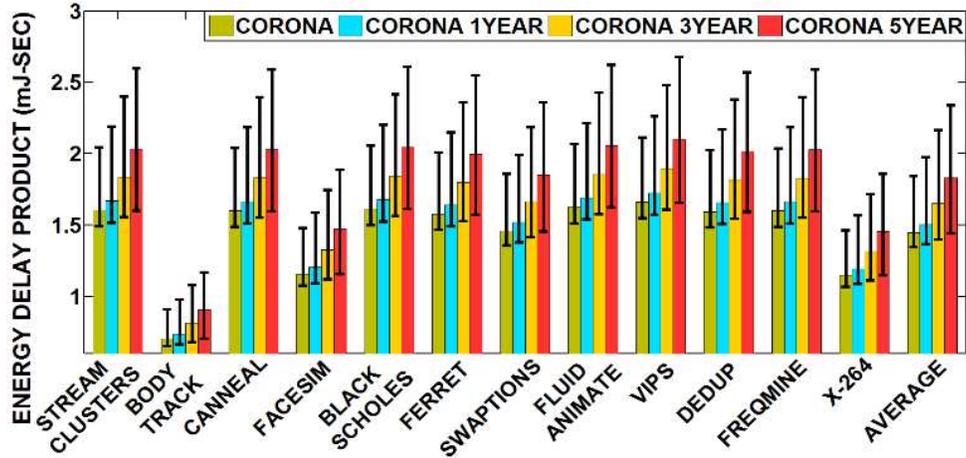


Figure 85 Worst-case signal power loss analysis of (a) Corona PNoC and (b) Clos PNoC, with 1 Year, 3 Years, and 5 Years of aging across 100 PV maps.

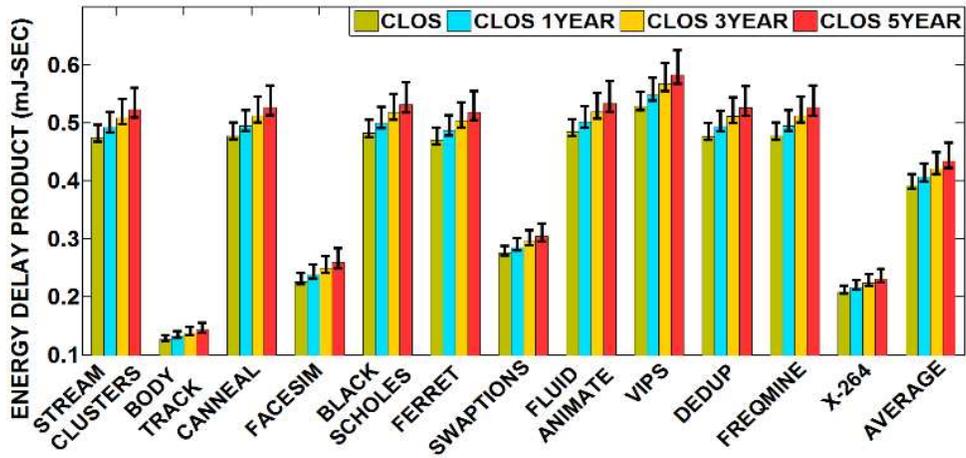
10.7.2. EXPERIMENT RESULTS

Our first set of experiments compares the worst-case signal losses of the baseline Corona and Clos PNoCs with their variants with 1 Year, 3 Years, and 5 Years of VBTI aging. We have performed this aging analysis across 100 PV maps as explained in Section 10.6.2. The presented results are averaged across the PV maps. Furthermore, as we are determining worst-case signal

loss for Corona and Clos PNoCs with VBTI aging, therefore we performed this analysis at the peak on-chip temperature, which is estimated to be 357 K [174].



(a)



(b)

Figure 86 EDP comparison of (a) Corona and (b) Clos PNoCs with 1 Year, 3 Years, and 5 Years of aging considering 100 process variation maps.

Utilizing the models presented in Section 10.6, we calculate the signal power loss at the last detector of the N_{WCPL} nodes of Corona and Clos PNoCs, which corresponds to the last MR detector of the cluster 64 and cluster 8 for the Corona and Clos PNoCs respectively. Figure 86(a) and (b)

compare the worst-case signal loss of baseline Corona and Clos PNoCs with three variants of these PNoCs that undergo 1 Year, 3 Years and 5 Years of VBTI aging. The confidence intervals represent the variation in signal loss across the 100 PV maps considered. From Figure 86(a), it can be observed that compared to their respective baselines, the Corona PNoC with 1 Year, 3 Year, and 5 years of VBTI aging has 2.8dB, 5.5dB, and 7.6dB higher signal losses, and the Clos PNoC has 1.1 dB, 2.1dB, and 2.6dB higher signal losses. The increase in resonance wavelength red shift ($\Delta\lambda_{RWRS}$) and degradation in Q-factor with VBTI aging in MRs leads to increase in MR loss factor (γ) (see Eq. (79)) and decrease in MR coupling factor (Φ) (see Eq. (78)), which ultimately increases signal losses in these PNoCs. Also, the increase in signal loss in the Corona PNoC with VBTI aging is on the higher side compared to the Clos PNoC. Corona has 16 \times higher number of MRs on its waveguides compared to the Clos PNoC, which in turn incurs higher signal losses on Corona's waveguides.

Figure 86(a) and (b) present detailed simulation results that quantify the energy-delay product (EDP) for the four configurations of Corona and Clos PNoCs respectively. Results are shown for twelve multi-threaded PARSEC benchmarks. From Fig. 7(a) it can be seen that on average, Corona PNoC with 1 Year, 3 Year, and 5 years of VBTI aging has 4.1%, 14.3%, and 26.8% and Clos PNoC has 3.7%, 7.5%, and 10.6% higher EDP compare to their respective baselines. Increase in worst-case signal loss with increase in VBTI aging (see Figure 85) contributes to an increase in the PNoCs laser power, which increases total laser energy consumption in these PNoCs. Additionally, VBTI aging in MRs has positive effects on MR trimming energy consumption, as MR aging incurs red shift in resonance wavelength which naturally reduces PV-induced blue shifts in MRs and reduces total trimming energy consumption

in the PNoCs. However, these trimming energy savings are relatively on the lower side compared to the increase in laser energy, which ultimately increase total energy and hence the EDP.

From the results presented in this section, we can summarize that in Corona and Clos PNoCs, VBTI aging in MRs increases signal losses by up to 7.6dB. Despite the decrease in tuning energy consumption of the Corona and Clos PNoCs with VBTI aging, the increase in their laser energy consumption increases EDP in these architectures by up to 26.8%. The signal loss and EDP increase due to VBTI aging are much lower in architectures optimized for physical-layouts such as the Clos PNoC, than in non-optimized architectures such as Corona. PNoC architectures with more MRs per waveguide (e.g., Corona) have higher VBTI aging degradation compared to PNoC architectures with less MRs per waveguide (e.g., Clos). Thus, to reduce aging effects in a PNoC, designers should reduce the number of MRs per waveguide and increase the number of these waveguides to maintain high bandwidth.

10.8. CONCLUSIONS

This chapter analyzed VBTI aging in MRs used in photonic interconnects, and the dependence of this aging on voltage bias and temperature. We presented an analytical model for trap generation on the MR core-cladding boundary with VBTI aging in MRs. We also consider the impact of process variations on aging. Our device-level results indicate that MR aging causes significant degradation in MR Q-factor and incurs notable resonance wavelength red shift. We extended our MR aging analysis to the system-level for the Corona and Clos PNoCs. The system-level analysis on these PNoCs clearly shows the damaging effects of MR aging, with worst signal loss increase by up to 7.6dB and EDP increase by up to 26.8%.

11. SOTERIA: EXPLOITING PROCESS VARIATIONS TO ENHANCE HARDWARE SECURITY WITH PHOTONIC NOC ARCHITECTURES

A Hardware Trojan in a PNoC can manipulate the electrical driving circuit of its MRs to cause the MRs to snoop data from the neighboring wavelength channels in a shared photonic waveguide. This introduces a serious security threat. This chapter presents a novel framework called *SOTERIA* that utilizes process variation based authentication signatures along with architecture-level enhancements to protect data in PNoC architectures from snooping attacks. Evaluation results indicate that our approach can significantly enhance the hardware security in DWDM-based PNoCs with minimal overheads of up to 10.6% in average latency and of up to 13.3% in energy-delay-product (EDP).

11.1. INTRODUCTION

To cope with the growing performance demands of modern Big Data and cloud computing applications, the complexity of hardware in modern chip-multiprocessors (CMPs) has increased. To reduce the hardware design time of these complex CMPs, third-party hardware IPs are frequently used. But these third party IPs can introduce security risks [175]- [176]. For instance, the presence of Hardware Trojans (HTs) in the third-party IPs can lead to leakage of critical and sensitive information from modern CMPs [177]. Thus, security researchers that have traditionally focused on software-level security are now increasingly interested in overcoming hardware-level security risks.

Many CMPs today use electrical networks-on-chip (ENoCs) [178] for inter-core communication. ENoCs use packet-switched network fabrics and routers to transfer data between

on-chip components [179]. Recent developments in silicon photonics have enabled the integration of photonic components and interconnects with CMOS circuits on a chip. Photonic NoCs (PNoCs) provide several prolific advantages over their metallic counterparts (i.e., ENoCs), including the ability to communicate at near light speed, larger bandwidth density, and lower dynamic power dissipation [180], [181]. These advantages motivate the use of PNoCs for inter-core communication in modern CMPs [21], [182].

Several PNoC architectures have been proposed to date (e.g., [11], [13], [183], [184], [185]). These architectures employ on-chip photonic links, each of which connects two or more gateway interfaces. A gateway interface (GI) connects the PNoC to a cluster of processing cores. Each photonic link comprises one or more photonic waveguides and each waveguide can support a large number of dense-wavelength-division-multiplexed (DWDM) wavelengths. A wavelength serves as a data signal carrier. Typically, multiple data signals are generated at a source GI in the electrical domain (as sequences of logical 1 and 0 voltage levels) which are modulated onto the multiple DWDM carrier wavelengths simultaneously, using a bank of modulator MRs at the source GI. The data-modulated carrier wavelengths traverse a link to a destination GI, where an array of detector MRs filter them and drop them on photodetectors to regenerate electrical data signals.

In general, each GI in a PNoC is able to send and receive data in the optical domain on all of the utilized carrier wavelengths [186]. Therefore, each GI has a bank of modulator MRs (i.e., modulator bank) and a bank of detector MRs (i.e., detector bank). Each MR in a bank resonates with and operates on a specific carrier wavelength. Thus, the excellent wavelength selectivity of MRs and DWDM capability of waveguides enable high bandwidth parallel data transfers in PNoCs.

Similar to CMPs with ENoCs, the CMPs with PNoCs are expected to use several third party IPs, and therefore, are vulnerable to security risks [187]- [188]. For instance, if the entire PNoC

used within a CMP is a third-party IP, then this PNoC with HTs within the control units of its GIs can snoop on packets in the network. These packets can be transferred to a malicious core (a core running a malicious program) in the CMP to determine sensitive information.

Unfortunately, MRs are especially susceptible to security threatening manipulations from HTs. In particular, *the MR tuning circuits that are essential for supporting data broadcasts and to counteract MR resonance shifts due to process variations (PV) make it easy for HTs to retune MRs and initiate snooping attacks.* To enable data broadcast in PNoCs, the tuning circuits of detector MRs partially detune them from their resonance wavelengths [12], [71], such that a significant portion of the photonic energy in the data-carrying wavelengths continues to propagate in the waveguide to be absorbed in the subsequent detector MRs. On the other hand, process variations (PV) cause resonance wavelength shifts in MRs [22]. Techniques to counteract PV-induced resonance shifts in MRs involve retuning the resonance wavelengths by using carrier injection/depletion or thermal tuning [21], implemented through MR tuning circuits. An HT in the GI can manipulate these tuning circuits of detector MRs to partially tune the detector MR to a passing wavelength in the waveguide, which enables snooping of the data that is modulated on the passing wavelength. *Such covert data snooping is a serious security risks in PNoCs.*

In this chapter, we present a framework that protects data from snooping attacks and improves hardware security in PNoCs. Our framework has low overhead and is easily implementable in any existing DWDM-based PNoC without major changes to the architecture. To the best of our knowledge, this is the first work that attempts to improve hardware security for PNoCs. Our novel contributions are:

- We analyze security risks in photonic devices and extend this analysis to link-level, to determine the impact of these risks on PNoCs;

- We propose a circuit-level PV-based security enhancement scheme that uses PV-based authentication signatures to protect data from snooping attacks in photonic waveguides;
- We propose an architecture-level reservation-assisted security enhancement scheme to improve security in DWDM-based PNoCs;
- We combine the circuit- and architecture-level schemes into a holistic framework called *SOTERIA*; and analyze it on the Firefly [12] and Flexishare [13] PNoC architectures.

11.2. RELATED WORK

Several prior works [188], [189], [190], [191] discuss the presence of security threats in ENoCs and have proposed solutions to mitigate them. In [188], a three-layer security system approach was presented by using data scrambling, packet certification, and node obfuscation to enable protection against data snooping attacks. In [189], a Hardware Trojan threat model was presented that covertly performs deep packet inspection and injects faults on links to develop a denial-of-service (DoS) attack. A symmetric-key based cryptography design was presented in [190] for securing the NoC. In [191], a framework was presented to use permanent keys and temporary session keys for NoC transfers between secure and non-secure cores. *However, no prior work has analyzed security risks in on-chip photonic devices and links; or considered the impact of these risks on PNoC architectures.*

Fabrication-induced process variations (PV) impact the cross-section, i.e., width and height, of photonic devices, such as MRs and waveguides. In MRs, PV causes resonance wavelength drifts, which can be counteracted by using device-level techniques such as thermal tuning or localized trimming [21]. Trimming can induce blue shifts in the resonance wavelengths of MRs using carrier injection into MRs, whereas thermal tuning can induce red shifts in MR resonances through

heating of MRs using integrated heaters. *To remedy PV, the use of device-level trimming/tuning techniques is inevitable; but their use also enables partial detuning of MRs that can be used to snoop data from a shared DWDM-based photonic waveguide.*

Our proposed framework in this chapter is novel as it enables security against snooping attacks in PNoCs for the first time. Our framework is network agnostic, mitigates PV, and has minimal overhead, while improving security for any DWDM-based PNoC architecture.

11.3. HARDWARE SECURITY CONCERNS IN PNOCS

11.3.1. DEVICE-LEVEL SECURITY CONCERNS

Process variation (PV) induced undesirable changes in MR widths and heights cause “shifts” in MR resonance wavelengths, which can be remedied using localized trimming and thermal tuning methods. The localized trimming method injects (or depletes) free carriers into (or from) the Si core of an MR using an electrical tuning circuit, which reduces (or increases) the MR’s refractive index owing to the electro-optic effect, thereby remedying the PV-induced red (or blue) shift in the MR’s resonance wavelength. In contrast, thermal tuning employs an integrated micro-heater to adjust the temperature and refractive index of an MR (owing to the thermo-optic effect) for PV remedy. Typically, the modulator MRs and detectors use the same electro-optic effect (i.e., carrier injection/depletion) implemented through the same electrical tuning circuit as used for localized trimming, to move in and out of resonance (i.e., switch ON/OFF) with a wavelength [18]. *A Hardware Trojan can manipulate this electrical tuning circuit, which may lead to malicious operation of modulator and detector MRs, as discussed next.*

Figure 87(a) shows the malicious operation of a modulator MR. A malicious modulator MR is partially tuned to a data-carrying wavelength (shown in purple) that is passing by in the

waveguide. The malicious modulator MR draws some power from the data-carrying wavelength, which can ultimately lead to data corruption as optical ‘1’s in the data can lose significant power to be altered into ‘0’s. Alternatively, a malicious detector (Figure 87(b)) can be *partially* tuned to a passing data-carrying wavelength, to filter only a small amount of its power and drop it on a photodetector for data duplication. This small amount of filtered power does not alter the data in the waveguide so that it continues to travel to its target detector for legitimate communication [71]. Thus, malicious detector MRs can snoop data from the waveguide without altering it, which is a major security threat in photonic links. Note that malicious modulator MRs only corrupt data (which can be detected) and do not covertly duplicate it, and are thus not a major security risk. Our analysis in Section 11.3.2 presents the impact of malicious modulator and detector MRs on photonic links.

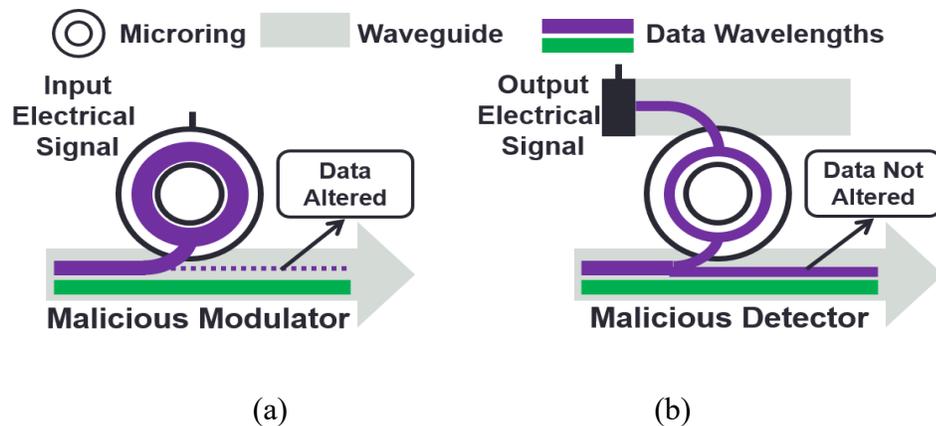


Figure 87 Impact of (a) malicious modulator MR, (b) malicious detector MR on data in DWDM-based photonic waveguides.

11.3.2. LINK-LEVEL SECURITY CONCERNS

Typically, a photonic link is comprised of one or more DWDM-based photonic waveguides. A DWDM-based photonic waveguide uses a modulator bank (a series of modulator MRs) at the

source GI and a detector bank (a series of detector MRs) at the destination GI. DWDM-based waveguides can be broadly classified into four types: single-writer-single-reader (SWSR), single-writer-multiple-reader (SWMR), multiple-writer-single-reader (MWSR), and multiple-writer-multiple-reader (MWMR). As SWSR, SWMR, and MWSR waveguides are subsets of an MWMR waveguide, and due to limited space, we restrict our link-level analysis to MWMR waveguides.

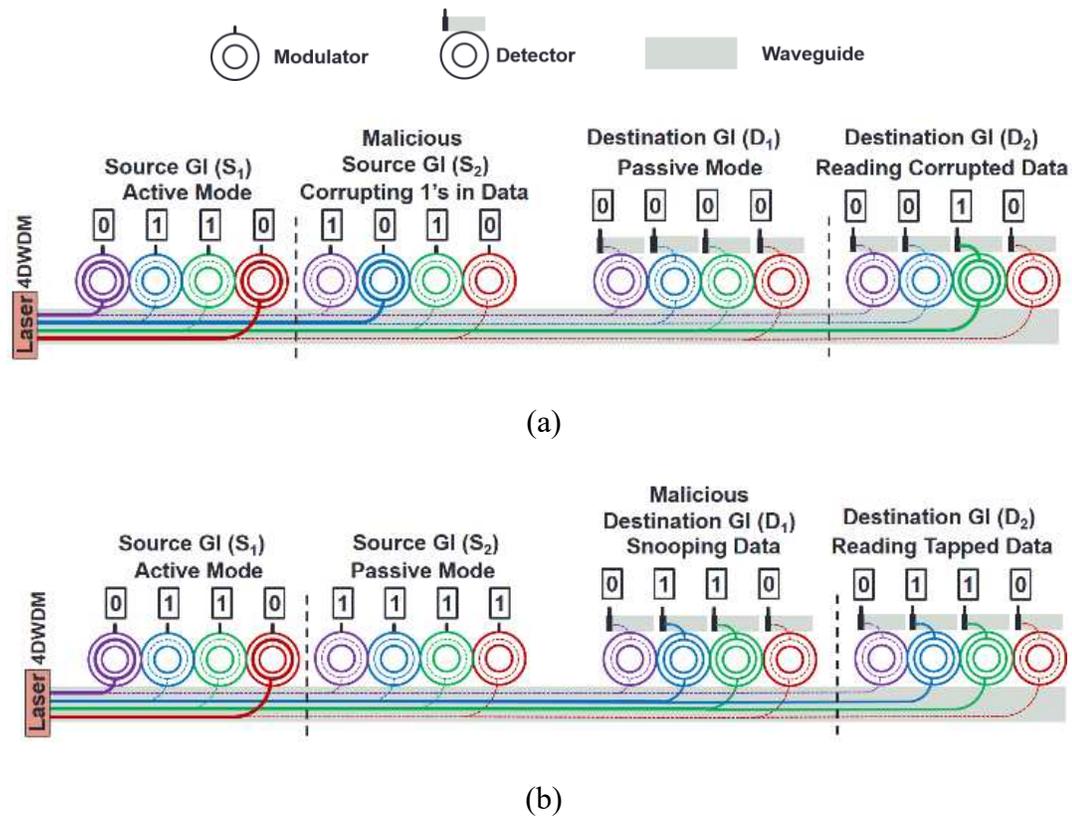


Figure 88 Impact of (a) malicious modulator (source) bank, (b) malicious detector bank on data in DWDM-based photonic waveguides.

An MWMR waveguide typically passes through multiple GIs, connecting the modulator banks of some GIs to the detector banks of the remaining GIs. Thus, in an MWMR waveguide, multiple GIs (referred to as source GIs) can send data using their modulator banks and multiple GIs (referred to as destination GIs) can receive (read) data using their detector banks. Figure 88

presents an example MWMMR waveguide with two source GIs and two destination GIs. Figure 88(a) and Figure 88(b), respectively, present the impact of malicious source and destination GIs on this MWMMR waveguide. In Figure 88(a), the modulator bank of source GI S_1 is sending data to the detector bank of destination GI D_2 . When source GI S_2 , which is in the communication path, becomes malicious with an HT in its control logic, it can manipulate its modular bank to modify the existing ‘1’s in the data to ‘0’s. This ultimately leads to data corruption. For example, in Figure 88(a), S_1 is supposed to send ‘0110’ to D_2 , but because of data corruption by malicious GI S_2 , ‘0010’ is received by D_2 . Nevertheless, this type of data corruption can be detected or even corrected using parity or error correction code (ECC) bits in the data. Thus, malicious source GIs do not cause major security risks in DWDM-based MWMMR waveguides.

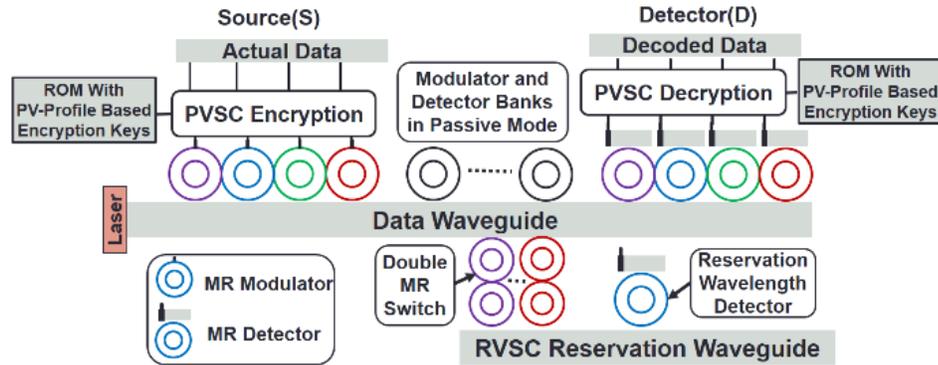


Figure 89 Overview of proposed *SOTERIA* framework that integrates a circuit-level PV-based security enhancement (*PVSC*) scheme and an architecture-level reservation-assisted security enhancement (*RVSC*) scheme.

Let us consider another scenario for the same data communication path (i.e., from S_1 to D_2). When destination GI D_1 , which is in the communication path, becomes malicious with an HT in its control logic, the detector bank of D_1 can be partially tuned to the utilized wavelength channels to snoop data. In the example shown in Figure 88(b), D_1 snoops ‘0110’ from the wavelength channels that are destined to D_2 . The snooped data from D_1 can be transferred to a malicious core

within the CMP to determine sensitive information. This type of snooping attack from malicious destination GIs is hard to detect, as it does not disrupt the intended communication among CMP cores. Therefore, there is a pressing need to address the security risks imposed by snooping GIs in DWDM-based PNoC architectures. To address this need, we propose a novel framework *SOTERIA* that improves hardware security in DWDM-based PNoC architectures.

11.4. SOTERIA FRAMEWORK: OVERVIEW

Our proposed multi-layer *SOTERIA* framework enables secure communication in DWDM-based PNoC architectures by integrating circuit-level and architecture-level enhancements. Figure 89 gives a high-level overview of this framework. The PV-based security enhancement (*PVSC*) scheme uses the PV profile of the destination GIs' detector MRs to encrypt data before it is transmitted via the photonic waveguide. This scheme is sufficient to protect data from snooping GIs, if they do not know about the target destination GI. With target destination GI information, however, a snooping GI can decipher the encrypted data. Many PNoC architectures (e.g., [14], [188]) use the same waveguide to transmit both the destination GI information and actual data, making them vulnerable to data snooping attacks despite using *PVSC*. To further enhance security for these PNoCs, we devise an architecture-level reservation-assisted security enhancement (*RVSC*) scheme that uses a secure reservation waveguide to avoid the stealing of destination GI information by snooping GIs. Next two sections present details of our *PVSC* and *RVSC* schemes.

11.5. PV-BASED SECURITY ENHANCEMENT

As discussed earlier (Section 11.3.2), malicious destination GIs can snoop data from a shared waveguide. One way of addressing this security concern is to use data encryption so that the malicious destination GIs cannot decipher the snooped data. For the encrypted data to be truly

undecipherable, the encryption key used for data encryption should be kept secret from the snooping GIs, which can be challenging as the identity of the snooping GIs in a PNoC is not known. Therefore, it becomes very difficult to decide whether or not to share the encryption key with a destination GI (that can be malicious) for data decryption. This conundrum can be resolved using a different key for every destination GI so that a key that is specific to a secure destination GI does not need to be shared with a malicious destination GI for decryption purpose. Moreover, to keep these destination specific keys secure, the malicious GIs in a PNoC must not be able to clone the algorithm (or method) used to generate these keys.

To generate unclonable encryption keys, our PV-based security (*PVSC*) scheme uses the PV profiles of the destination GIs' detector MRs. As discussed in [22], PV induces random shifts in the resonance wavelengths of the MRs used in a PNoC. These resonance shifts can be in the range from -3nm to 3nm [22]. The MRs that belong to different GIs in a PNoC have different PV profiles. In fact, the MRs that belong to different MR banks of the same GI also have different PV profiles. Due to their random nature, these MR PV profiles cannot be cloned by the malicious GIs, which makes the encryption keys generated using these PV profiles truly unclonable. Using the PV profiles of detector MRs, *PVSC* can generate a unique encryption key for each detector bank of every MWMMR waveguide in a PNoC.

Our *PVSC* scheme generates encryption keys during the testing phase of the CMP chip, by using a dithering signal based in-situ method [192] to generate an anti-symmetric analog error signal for each detector MR of every detector bank that is proportional to the PV-induced resonance shift in the detector MR. Then, it converts the analog error signal into a 64-bit digital signal. Thus, a 64-bit digital error signal is generated for every detector MR of each detector bank. We consider 64 DWDM wavelengths per waveguide, and hence, we have 64 detector MRs in every detector

bank and 64 modulator MRs in every modulator bank. For each detector bank, our *PVSC* scheme XORs the 64 digital error signals (of 64 bits each) from each of the 64 detector MRs to create a unique 64-bit encryption key. Note that our *PVSC* scheme also uses the same anti-symmetric error signals to control the carrier injection and heating of the MRs to remedy the PV-induced shifts in their resonances.

To understand how the 64-bit encryption key is utilized to encrypt data in photonic links, consider Figure 90 which depicts an example photonic link that has one MWMMR waveguide and connects the modulator banks of two source GIs (S_1 and S_2) with the detector banks of two destination GIs (D_1 and D_2). As there are two destination GIs on this link, *PVSC* creates two 64-bit encryption keys corresponding to them, and stores them at the source GIs. When data is to be transmitted by a source GI, the key for the appropriate destination is used to encrypt data at the flit-level granularity, by performing an XOR between the key and the data flit. This requires that the size of an encryption key match the data flit size. We consider the size of data flits to be 512 bits. Therefore, the 64-bit encryption key is appended eight times to generate a 512-bit encryption key. In Figure 90, every source GI stores two 512-bit encryption keys (for destination GIs D_1 and D_2) in its local ROM, whereas every destination GI stores only its corresponding 512-bit key in its ROM. Note that we store the 512-bit keys instead of the 64-bit keys as this eliminates the latency overhead of affixing 64-bit keys to generate 512-bit keys, at the cost of a reasonable area/energy overhead in the ROM. As an example, if S_1 wants to send a data flit to D_2 , then S_1 first accesses the 512-bit encryption key corresponding to D_2 from its local ROM and XORs the data flit with this key in one cycle, and then transmits the encrypted data flit over the link. As the link employs only one waveguide with 64 DWDM wavelengths, therefore, the encrypted 512-bit data flit is transferred on the link to D_2 in eight cycles. At D_2 , the data flit is decrypted by XORing it with the

512-bit key corresponding to D_2 from the local ROM. In this scheme, even if D_1 snoops the data intended for D_2 , it cannot decipher the data as it does not have access to the correct key (corresponding to D_2) for decryption. Thus, our *PVSC* encryption scheme protects data against snooping attacks in DWDM-based PNoCs.

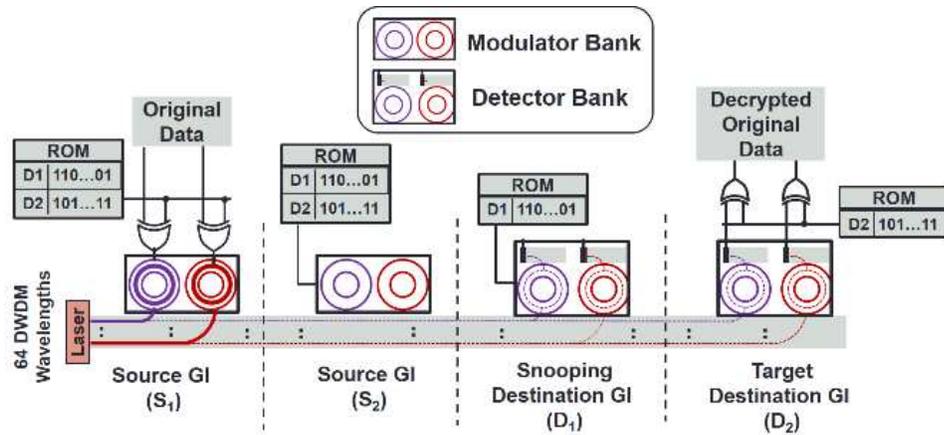


Figure 90 Overview of proposed PV-based security enhancement scheme.

Limitations of PVSC: The *PVSC* scheme can protect data from being deciphered by a snooping GI, if the following two conditions about the underlying PNoC architecture hold true: (i) the snooping GI does not know the target destination GI for the snooped data, (ii) the snooping GI cannot access the encryption key corresponding to the target destination GI. As discussed earlier, an encryption key is stored only at all source GIs and at the corresponding destination GI, which makes it physically inaccessible to a snooping destination GI. However, if more than one GIs in a PNoC are compromised due to HTs in their control units and if these HTs launch a coordinated snooping attack, then it may be possible for the snooping GI to access the encryption key corresponding to the target destination GI.

For instance, consider the photonic link in Figure 90. If both S_1 and D_1 are compromised, then the HT in S_1 's control unit can access the encryption keys corresponding to both D_1 and D_2

from its ROM and transfer them to a malicious core (a core running a malicious program). Moreover, the HT in D_1 's control unit can snoop the data intended for D_2 and transfer it to the malicious core. Thus, the malicious core may have access to the snooped data as well as the encryption keys stored at the source GIs. Nevertheless, accessing the encryption keys stored at the source GIs is not sufficient for the malicious GI (or core) to decipher the snooped data. This is because the compromised ROM typically has multiple encryption keys corresponding to multiple destination GIs, and choosing a correct key that can decipher data requires the knowledge of the target destination GI. Thus, our *PVSC* encryption scheme can secure data communication in PNoCs as long as the malicious GIs (or cores) do not know the target destinations of the snooped data.

Unfortunately, many PNoC architectures, e.g., [14], [188], that employ photonic links with multiple destination GIs utilize the same waveguide to transmit both the target destination information and actual data. In these PNoCs, if a malicious GI manages to tap the target destination information from the shared waveguide, then it can access the correct encryption key from the compromised ROM to decipher the snooped data. Thus, there is a need to conceal the target destination information from malicious GIs (cores). This motivates us to propose an architecture-level solution, as discussed next.

11.6. RESERVATION-ASSISTED SECURITY ENHANCEMENT

In PNoCs that use photonic links with multiple destination GIs, data is typically transferred in two time-division-multiplexed (TDM) slots called reservation slot and data slot [14], [188]. To minimize photonic hardware, PNoCs use the same waveguide to transfer both slots, as shown in Figure 91(a). To enable reservation of the waveguide, each destination is assigned a reservation

selection wavelength. In Figure 91(a), λ_1 and λ_2 are the reservation selection wavelengths corresponding to destination GIs D_1 and D_2 , respectively. Ideally, when a destination GI detects its reservation selection wavelength in the reservation slot, it switches ON its detector bank to receive data in the next data slot. But in the presence of an HT, a malicious GI can snoop signals from the reservation slot using the same detector bank that is used for data reception. For example, in Figure 91(a), malicious GI D_1 is using one of its detectors to snoop λ_2 from the reservation slot. By snooping λ_2 , D_1 can identify that the data it will snoop in the subsequent data slot will be intended for destination D_2 . Thus, D_1 can now choose the correct encryption key from the compromised ROM to decipher its snooped data.

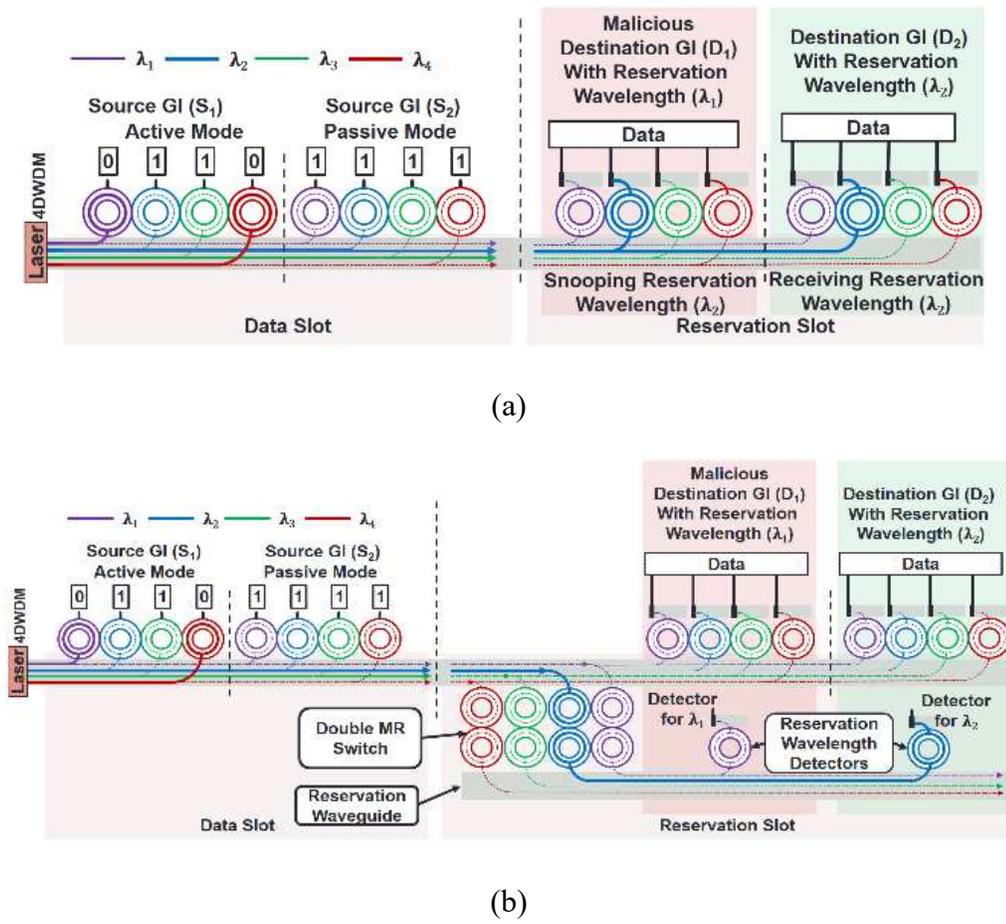


Figure 91 Reservation-assisted data transmission in DWDM-based photonic waveguides (a) without *RVSC*, (b) with *RVSC*.

To address this security risk, we propose an architecture-level reservation-assisted security enhancement (*RVSC*) scheme. In *RVSC*, we add a reservation waveguide, whose main function is to carry reservation slots, whereas the data waveguide carries data slots. We use double MRs to switch the signals of reservation slots from the data waveguide to the reservation waveguide, as shown in Figure 91(b). Double MRs are used instead of single MRs for switching to ensure that the switched signals do not reverse their propagation direction after switching [32]. Compared to single MRs, double MRs also have lower signal loss due to steeper roll-off of their filter response.

The double MRs are switched ON only when the photonic link is in a reservation slot, otherwise they are switched OFF to let the signals of the data slot pass by in the data waveguide. Furthermore, in *RVSC*, each destination GI has only one detector on the reservation waveguide, which corresponds to its receiver selection wavelength. For example, in Figure 91(b), D_1 and D_2 will have detectors corresponding to their reservation selection wavelengths λ_1 and λ_2 , respectively, on the reservation waveguide. This makes it difficult for the malicious GI D_1 to snoop λ_2 from the reservation slot as shown in Figure 91(b), as D_1 does not have a detector corresponding to λ_2 on the reservation waveguide. However, the HT in D_1 's control unit may still attempt to snoop other reservation wavelengths (e.g., λ_2) in the reservation slot by retuning D_1 's λ_1 detector. But succeeding in these attempts would require the HT to perfect the timing and target wavelength of its snooping attack, which is very difficult due to the large number of utilized reservation wavelengths. Thus, D_1 cannot identify the correct encryption key to decipher the snooped data.

In summary, *RVSC* enhances security in PNoCs by protecting data from snooping attacks, even if the encryption keys used for data encryption are compromised. To implement *RVSC* on a data waveguide with multiple destination GIs, we need to add a reservation waveguide with multiple detector MRs, where each detector MR corresponds to a destination GI. A group of double

MRs, each of which corresponds to a reservation selection wavelength available in the waveguide, is also needed to switch the wavelength signals of reservation slots from the data waveguide to the reservation waveguide. The introduction of the additional reservation waveguide and the group of double MRs increases signal loss and laser power. We account for this overhead in our PNoC architecture level analysis.

11.7. IMPLEMENTING *SOTERIA* FRAMEWORK ON PNOCS

We characterize the impact of *SOTERIA* on two popular PNoC architectures: Firefly [12] and Flexishare [13], both of which use DWDM-based photonic waveguides for data communication. We consider Firefly PNoC with 8×8 SWMR crossbar [12] and a Flexishare PNoC with 32×32 MWMR crossbar [13] with 2-pass token stream arbitration. We adapt the analytical equations from [32] to model the signal power loss and required laser power in the *SOTERIA*-enhanced Firefly and Flexishare PNoCs. At each source and destination GI of the *SOTERIA*-enhanced Firefly and Flexishare PNoCs, XOR gates are required to enable parallel encryption and decryption of 512-bit data flits. We consider a 1 cycle delay overhead for encryption and decryption of every data flit. The overall laser power and delay overheads for both PNoCs are quantified in the results section.

Firefly PNoC: Firefly PNoC [12], for a 256-core system, has 8 clusters (C1-C8) with 32 cores in each cluster. Firefly uses reservation-assisted SWMR data channels in its 8×8 crossbar for inter-cluster communication. Each data channel consists of 8 SWMR waveguides, with 64 DWDM wavelengths in each waveguide. To integrate *SOTERIA* with Firefly PNoC, we added a reservation waveguide to every SWMR channel. This reservation waveguide has 7 detector MRs to detect reservation selection wavelengths corresponding to 7 destination GIs. Furthermore, 64 double

MRs (corresponding to 64 DWDM wavelengths) are used at each reservation waveguide to implement *PVSC*. To enable *PVSC*, each source GI has a ROM with seven entries of 512 bits each to store seven 512-bit encryption keys corresponding to seven destination GIs. In addition, each destination GI requires a 512-bit ROM to store its own encryption key.

Flexishare PNoC: We also integrate *SOTERIA* with the Flexishare PNoC architecture [13] with 256 cores. We considered a 64-radix 64-cluster Flexishare PNoC with four cores in each cluster and 32 data channels for inter-cluster communication. Each data channel has four MWMM waveguides with each having 64 DWDM wavelengths. In *SOTERIA*-enhanced Flexishare, we added a reservation waveguide to each MWMM channel. Each reservation waveguide has 16 detector MRs to detect reservation selection wavelengths corresponding to 16 destination GIs. To enable *PVSC*, each source GI requires a ROM with 16 entries of 512 bits each to store the encryption keys, whereas each destination GI requires a 512-bit ROM.

Modeling PV of MR Devices in Firefly and Flexishare: Similar to [32], we adapt the VARIUS tool [20] to model random and systematic die-to-die (D2D) as well as within-die (WID) process variations in MRs for the Firefly and Flexishare PNoCs. We consider a 256-core chip with die size 400mm^2 at a 22nm process node. For this die-size we consider a WID standard deviation (σ_{WID}) of 0.61nm [22] and D2D standard deviation (σ_{D2D}) of 1.01 nm [22]. We also consider a density (α) of 0.5 [22] for this die size. With these parameters, we use VARIUS to generate 100 PV maps, each containing over 1 million points indicating the PV-induced resonance shift of MRs. The total number of points selected from these maps equal the number of MRs used in the Firefly and Flexishare PNoC architectures.

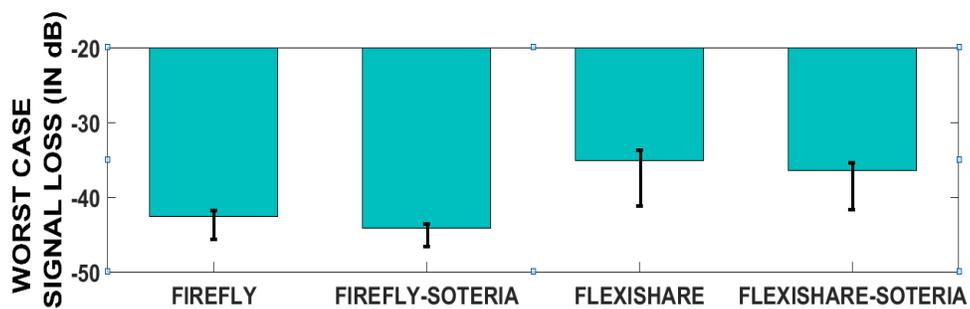
11.8. EXPERIMENTS

11.8.1. EXPERIMENT SETUP

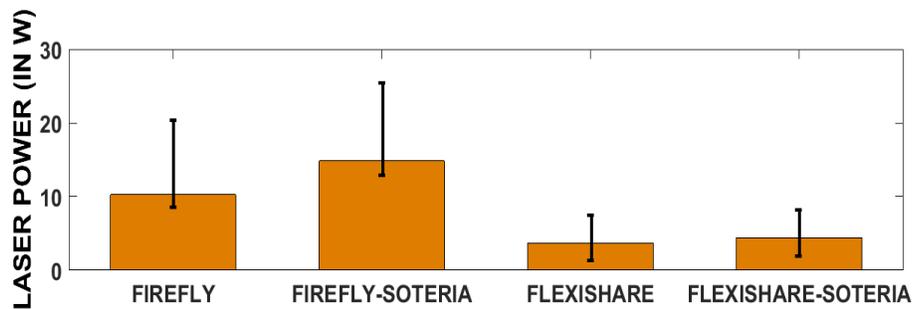
To evaluate our proposed *SOTERIA (PVSC+RVSC)* security enhancement framework for DWDM-based PNoCs, we integrate it with the Firefly [12] and Flexishare [13] PNoCs, as explained in Section 11.7. We modeled and performed simulation based analysis of the *SOTERIA*-enhanced Firefly and Flexishare PNoCs using a cycle-accurate SystemC based NoC simulator, for a 256-core single-chip architecture at 22nm. We validated the simulator in terms of power dissipation and energy consumption based on results obtained from the DSENT tool [75]. We used real-world traffic from the PARSEC benchmark suite [43]. GEM5 full-system simulation [72] of parallelized PARSEC applications was used to generate traces that were fed into our NoC simulator. We set a “warmup” period of 100 million instructions and then captured traces for the subsequent 1 billion instructions. These traces are extracted from parallel regions of execution of PARSEC applications. We performed geometric calculations for a 20mm×20mm chip size, to determine lengths of SWMR and MWMR waveguides in Firefly and Flexishare. Based on this analysis, we estimated the time needed for light to travel from the first to the last node as 8 cycles at 5 GHz clock frequency [25]. We use a 512-bit packet size, as advocated in the Firefly and Flexishare PNoCs.

The static and dynamic energy consumption values for electrical routers and concentrators in Firefly and Flexishare PNoCs are based on results from DSENT [75]. We model and consider the area, power, and performance overheads for our framework implemented with the Firefly and Flexishare PNoCs as follows. *SOTERIA* with Firefly and Flexishare PNoCs has an electrical area overhead of 12.7mm² and 3.4mm², respectively, and power overhead of 0.44W and 0.36W, respectively, using gate-level analysis and CACTI 6.5 [114] tool for memory and buffers. The

photonic area of Firefly and Flexishare PNoCs is 19.83mm^2 and 5.2mm^2 , respectively, based on the physical dimensions [104] of their waveguides, MRs, and splitters. For energy consumption of photonic devices, we adapt model parameters from recent work [25], [74], with 0.42pJ/bit for every modulation and detection event and 0.18pJ/bit for the tuning circuits of modulators and photodetectors. The MR trimming power is $130\mu\text{W/nm}$ [21] for current injection (to remedy red shifts) and tuning power is $240\mu\text{W/nm}$ [21] for heating (to remedy blue shifts).



(a)



(b)

Figure 92 Comparison of (a) worst-case signal loss and (b) laser power dissipation of *SOTERIA* framework on Firefly and Flexishare PNoCs with their respective baselines considering 100 process variation maps.

11.8.2. OVERHEAD ANALYSIS OF SOTERIA ON PNOCS

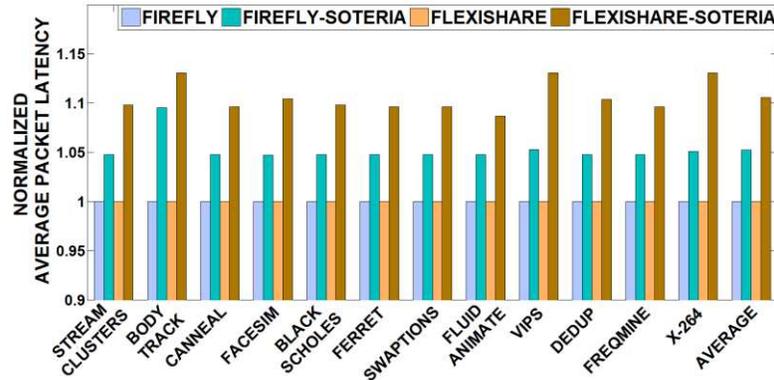
Our first set of experiments compare the baseline (without any security enhancements) Firefly and Flexishare PNoCs with their *SOTERIA* enhanced variants. From Section 11.7, all 8

SWMR waveguide groups of the Firefly PNoC and all 32 MWMR waveguide groups of the Flexishare PNoC are equipped with *PVSC* encryption/decryption and reservation waveguides for the *RVSC* scheme.

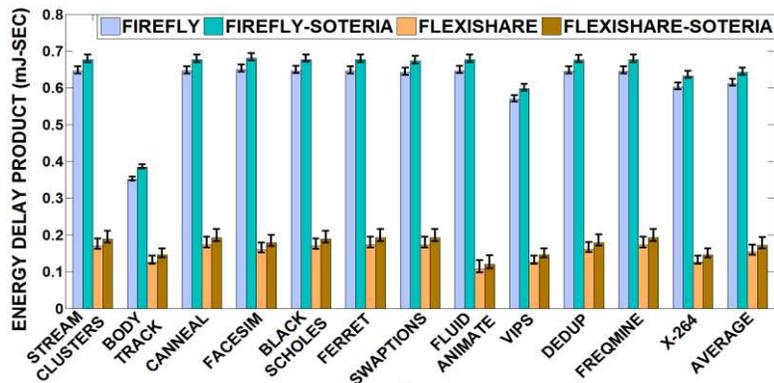
We adapt the analytical models from [32] to calculate the total signal loss at the detectors of the worst-case power loss node (N_{WCPL}), which corresponds to router C4R0 for the Firefly PNoC [12] and node R₆₃ for the Flexishare PNoC [13]. Figure 92(a) summarizes the worst-case signal loss results for the baseline and *SOTERIA* configurations for the two PNoC architectures. From the figure, Firefly PNoC with *SOTERIA* increases loss by 1.6dB and Flexishare PNoC with *SOTERIA* increases loss by 1.2dB on average, compared to their respective baselines. Compared to the baseline PNoCs that have no single or double MRs to switch the signals of the reservation slots, the double MRs used in the *SOTERIA*-enhanced PNoCs to switch the wavelength signals of the reservation slots increase through losses in the waveguides, which ultimately increases the worst-case signal losses in the *SOTERIA*-enhanced PNoCs. Using the worst-case signal losses shown in Figure 92(a), we determine the total photonic laser power and corresponding electrical laser power (using laser wall-plug efficiency of 3% [193]) for the baseline and *SOTERIA*-enhanced variants of Firefly and Flexishare PNoCs, shown in Figure 92(b). From this figure, the Firefly and Flexishare PNoCs with *SOTERIA* have laser power overheads of 44.7% and 31.40% on average, compared to their baselines.

Figure 93 presents detailed simulation results that quantify the average packet latency and energy-delay product (EDP) for the two configurations of the Firefly and Flexishare PNoCs. Results are shown for twelve multi-threaded PARSEC benchmarks. From Figure 93(a), Firefly with *SOTERIA* has 5.2% and Flexishare with *SOTERIA* has 10.6% higher latency on average

compared to their respective baselines. The additional delay due to encryption and decryption of data (Section 11.7) with *PVSC* contributes to the increase in average latency.



(a)



(b)

Figure 93 (a) normalized average latency and (b) energy-delay product (EDP) comparison between different variants of Firefly and Flexishare PNoCs that include their baselines and their variant with *SOTERIA* framework, for PARSEC benchmarks. Latency results are normalized with their respective baseline architecture results.

From the results for EDP shown in Figure 93(b), Firefly with *SOTERIA* has 4.9% and Flexishare with *SOTERIA* has 13.3% higher EDP on average compared to their respective baselines. Increase in EDP for the *SOTERIA*-enhanced PNoCs is not only due to the increase in their average packet latency, but also due to the presence of additional *RVSC* reservation

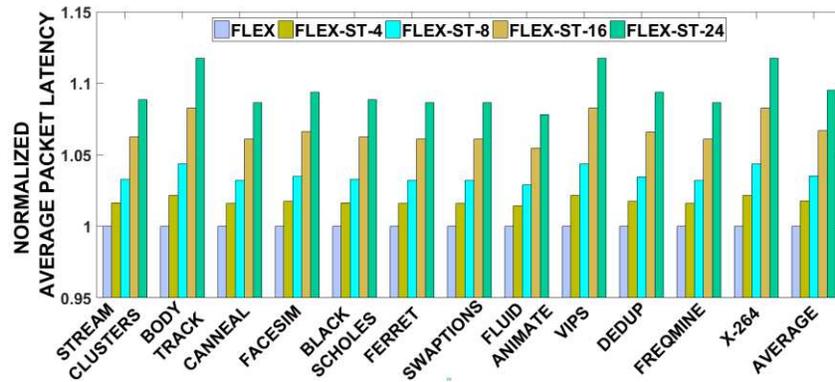
waveguides, which increases the required photonic hardware (e.g., more number of MRs) in the *SOTERIA*-enhanced PNoCs. This in turn increases static energy consumption (i.e., laser energy and trimming/tuning energy), ultimately increasing the EDP. From the results presented in this section, we can conclude that our *SOTERIA* framework improves hardware security in PNoCs at the cost of additional laser power, average latency, and EDP overheads.

11.8.3. ANALYSIS OF OVERHEAD SENSITIVITY

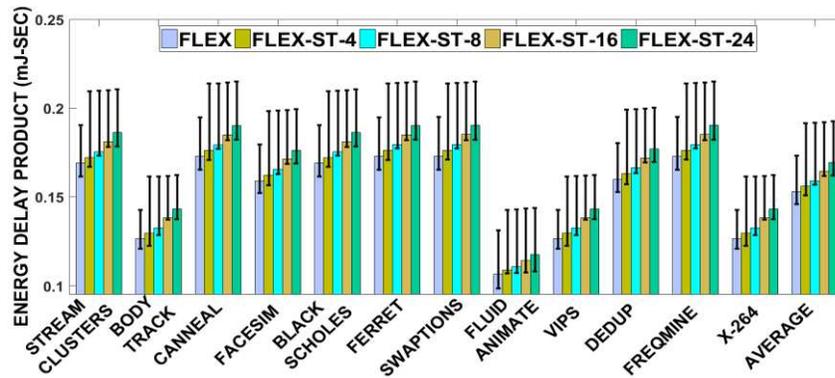
Our last set of evaluations explore how the overhead of *SOTERIA* changes with varying levels of security in the network. Typically, in a manycore system, only a certain portion of the data that contains sensitive information (i.e., keys) and only a certain number of communication links need to be secure. Therefore, for our analysis in this section, instead of securing all data channels of the Flexishare PNoC, we secure only a certain number channels using *SOTERIA*. Out of the total 32 MWMMR channels in the Flexishare PNoC, we secure 4 (FLEX-ST-4), 8 (FLEX-ST-8), 16 (FLEX-ST-16), and 24 (FLEX-ST-24) channels, and evaluate the average packet latency and EDP for these variants of the *SOTERIA*-enhanced Flexishare PNoC.

In Figure 94, we present average packet latency and EDP values for the five *SOTERIA*-enhanced configurations of the Flexishare PNoC. From Figure 94(a), FLEX-ST-4, FLEX-ST-8, FLEX-ST-16, and FLEX-ST-24 have 1.8%, 3.5%, 6.7%, and 9.5% higher latency on average compared to the baseline Flexishare. Increase in number of *SOTERIA* enhanced MWMMR waveguides increases number of packets that are transferred through the *PVSC* encryption scheme, which contributes to the increase in average packet latency across these variants. From the results for EDP shown in Figure 94(b), FLEX-ST-4, FLEX-ST-8, FLEX-ST-16, and FLEX-ST-24 have 2%, 4%, 7.6%, and 10.8% higher EDP on average compared to the baseline Flexishare. EDP in

Flexishare PNoC increases with increase in number of *SOTERIA* enhanced MWMR waveguides. Increase in average packet latency and signal loss due to the higher number of reservation waveguides and double MRs increase overall EDP across these variants.



(a)



(b)

Figure 94 (a) normalized latency and (b) energy-delay product (EDP) comparison between Flexishare baseline and Flexishare with 4, 8, 16, and 24 *SOTERIA* enhanced MWMR waveguide groups, for PARSEC benchmarks. Latency results are normalized to the baseline Flexishare results.

11.8.4. SUMMARY OF RESULTS AND OBSERVATIONS

From the results in the previous subsections, it can be concluded that our proposed *SOTERIA* framework secures data during unicast communications in PNoC architectures from snooping

attacks by leveraging the benefits of our circuit-level *PVSC* and architecture-level *RVSC* techniques. *SOTERIA*-enhanced PNoCs incur minimal overheads of up to 10.6% and as low as 1.8% in average packet latency and up to 13.3% and as low as 2% in EDP compared to the baseline insecure PNoCs.

11.9. CONCLUSIONS

We presented a novel security enhancement framework called *SOTERIA* that secures data during unicast communications in DWDM-based PNoC architectures from snooping attacks. Our proposed *SOTERIA* framework shows interesting trade-offs between security, performance, and energy overhead for the Firefly and Flexishare PNoC architectures. Our analysis shows that *SOTERIA* enables hardware security in crossbar based PNoCs with minimal overheads of up to 10.6% in average latency and of up to 13.3% in EDP compared to the baseline PNoCs. Thus, *SOTERIA* represents an attractive solution to enhance hardware security in emerging DWDM-based PNoCs. In the future, we plan to extend our *SOTERIA* framework to enhance data security during broadcast and multicast communications in PNoCs.

12. CONCLUSION AND FUTURE WORK SUGGESTIONS

12.1. RESEARCH CONCLUSION

We addressed several design challenges faced by PNoC architectures by proposing a framework which employs layer-specific solutions and cross-layer solutions that combine enhancements at the system-level, architecture-level, circuit-level, and device-level. Our proposed cross-layer framework utilizes various (i) system-level solutions such as application scheduling and thread migration, (ii) architectural-solutions such as application-specific, reconfigurable, and security-aware PNoC architecture designs, and (iii) device- and circuit-level solutions such as encoding and MR assignment, to optimize PNoC performance, energy efficiency, and reliability. Experimental results for our proposed cross-layer framework validate and motivate its deployment in future PNoC architectures, because it demonstrates significant improvement in energy efficiency with extensibility to adapt to new PNoC design concerns, such as crosstalk, thermal variations, process variations, security, and aging effects. Therefore, our proposed cross-layer framework has the potential to be applied as a general strategy for performance, reliability, power, and security management on DWDM-based PNoC architectures.

Our first contribution is the development of the *SwiftNoC* photonic NoC architecture which is an improved version of the UltraNoC architecture, with more efficient channel sharing among cores with an aggressive concurrent token stream-based arbitration strategy and more efficient multicast support. *SwiftNoC* supports the ability to dynamically transfer bandwidth between clusters of cores and to re-prioritize multiple co-running applications to further improve channel utilization and adapt to time-varying application performance goals. *SwiftNoC* improves

throughput by up to 25.4× while reducing latency by up to 72.4% and EPB by up to 95% over other state-of-the-art solutions. *SwiftNoC* also scales well with increasing chip core counts.

Our next contribution is the application-specific *BiGNoC* architecture that features master-servant clusters with efficient utilization of SWMR and MWSR waveguides to improve performance while executing large-scale data analytics applications. *BiGNoC* exploits efficient multicasting in photonic waveguides to achieve high data rates. In particular, we showed how the *BiGNoC-HET* variant of *BiGNoC*, improves performance due to improved photonic channel utilization and its ability to adapt to time-varying application performance goals while co-running multiple large-scale data analytics applications. *BiGNoC-HET* improves throughput by up to 9.9×, packet latency by up to 88%, and energy-per-bit by up to 98% over traditional EMesh, broadcast-optimized EMesh, and state-of-the-art photonic NoC architectures. These results corroborate the excellent capabilities of our proposed *BiGNoC* architecture towards executing large-scale data analytics applications.

Heterodyne crosstalk mitigation techniques are presented in this dissertation for the reduction of crosstalk noise in the detectors of DWDM-based PNoC architectures with crossbar topologies. These techniques (PCTM5B, PCTM6B, WSP, PICO, HYDRA) show interesting trade-offs between reliability, performance, and energy overheads across three different crossbar-based PNoC architectures. Our simulation based analysis shows that the proposed heterodyne crosstalk mitigation techniques improves worst-case OSNR by up to 5.3× compared to the baseline architectures. Thus, our proposed techniques are attractive solutions to enhance reliability in emerging DWDM-based PNoCs.

We presented the *IHDTM* and *LIBRA* frameworks that combine novel dynamic thermal management mechanisms for the reduction of maximum on-chip temperature and approaches for

the conservation of trimming and tuning power of MRs in DWDM-based PNoC architectures. These techniques (*TPMA* and on-chip islands at the device-level, *VADTM* and *TATM* at the system-level) constitute hybrid reactive-proactive management frameworks which demonstrated interesting trade-offs between performance and power/energy across two different state-of-the-art crossbar-based PNoC architectures. Our experimental analysis on state-of-the-art PNoC architectures has shown that the proposed frameworks can notably conserve total power by up to 61.3% (trimming and tuning power by up to 76.2%) and total energy by up to 57.3%.

VBTI aging in the MRs used in photonic interconnects, and the dependence of this aging on voltage bias and temperature was also analyzed in this dissertation. We presented an analytical model for trap generation on the MR core-cladding boundary with VBTI aging in MRs. We also considered the impact of process variations on aging. Our device-level results indicate that MR aging causes significant degradation in MR Q-factor and incurs notable resonance wavelength red shift. We extended our MR aging analysis to the system-level for two crossbar-based PNoCs. The system-level analysis on these PNoCs clearly shows the damaging effects of MR aging with a worst-case signal loss increase of up to 7.6dB and EDP increase of up to 26.8%.

Lastly, a novel security enhancement framework called *SOTERIA* that secures data during unicast communications in DWDM-based PNoC architectures from snooping attacks was also presented in this dissertation. The *SOTERIA* framework shows interesting trade-offs between security, performance, and energy overheads for DWDM-based PNoC architectures. Our analysis shows that *SOTERIA* enables hardware security in crossbar based PNoCs with minimal overheads of up to 10.6% in average latency and up to 13.3% in EDP compared to the baseline PNoCs. Thus, *SOTERIA* represents an attractive solution to enhance hardware security in emerging DWDM-based PNoCs.

12.2. SUGGESTION FOR FUTURE WORKS

PNoC architecture design will continue to face new challenges as large number of photonic devices are expected to be integrated on CMPs in the near future. Therefore, we envision the following likely directions for future work:

- *Fault Tolerant PNoC*: Faults are inevitable not only in electrical interconnects, but also in photonic interconnects. Faults in PNoCs include, photonic waveguide faults, MR faults, and splitter faults. There is a need to explore these faults in PNoCs and novel strategies are needed to reduce fault-induced performance penalties. A cross-layer approach combining architectural-level enhancements and system-level application scheduling will likely also be beneficial.
- *Aging in MRs*: In this dissertation, we already explored VBTI aging in MR's PN junctions (see chapter 10) and analyzed its impact on PNoC architectures. In addition to VBTI, MRs are prone to other aging mechanisms such as hot carrier injection (possible in PN junction of an MR) however these aging scenarios have not been explored yet.
- *Hardware Security in PNoCs*: Security is expected to be a critical concern in CMPs that use DWDM-based PNoCs for inter-core communication. Mechanisms to mitigate snoop-based attacks on PNoCs are already discussed in this dissertation (see chapter 11). Furthermore, novel strategies are needed to mitigate snoop-based attacks in multicast- and broadcast-enabled PNoC architectures. However, PNoCs are also vulnerable to Denial-of-service (DoS) based attacks and data corruption based attacks. Therefore, solutions are needed to reduce the aforementioned security risks in PNoCs to further enhance their hardware security.

BIBLIOGRAPHY

- [1] D. Pham, S. Asano, M. Bolliger, M. Day, H. Hofstee, C. Johns, J. Kahle, A. Kameyama, J. Keaty, Y. Masubuchi, M. Riley, D. Shippy, D. Stasiak, M. Suzuki, M. Wang, J. Warnock, S. Weitzel, D. Wendel, T. Yamazaki, and K. Yazawa, "The design and implementation of a first-generation CELL processor," in *Proceedings of Solid-State Circuits Conference*, 2005.
- [2] S. Pasricha and N. Dutt, "On-Chip Communication Architectures," *Morgan Kaufman*, pp. ISBN 978-0-12-373892-9, 2008.
- [3] S. Vangal, J. Howard, G. Ruhl, S. Dighe, H. Wilson, J. Tschanz, D. Finan, P. Iyer, A. Singh, T. Jacob, S. Jain, S. Venkataraman, Y. Hoskote, and N. Borkar, "An 80-Tile 1.28 TFLOPS Network-on-Chip in 65 nm CMOS," in *Proceedings of IEEE International Solid State Circuits Conference*, 2007.
- [4] Mellonox Corporation, "Mellonox multicore processors," <http://www.tilera.com/products/processors>.
- [5] Intel Corporation, "Intel Xeon Phi™ Processor 7290F," http://ark.intel.com/products/95831/Intel-Xeon-Phi-Processor-7290F-16GB-1_50-GHz-72-core, 2016.
- [6] Kalray Inc., "Kalray Bostan MPPA2 256-core processor," <http://www.kalrayinc.com/tag/bostan/>, 2015.

- [7] L. Zhou and A. K. Kodi, "PROBE: Prediction-based optical bandwidth scaling for energy-efficient NoCs," in *Proceedings of IEEE/ACM International Symposium on Networks-on-Chip*, 2013.
- [8] D. A. B. Miller, "Device requirements for optical interconnects to silicon chips," in *Proceedings of IEEE, Special Issue on Silicon Photonics*, vol. 97, no. 7, pp. 1166-1185, 2009.
- [9] S. Bahirat and S. Pasricha, "METEOR: Hybrid Photonic Ring-Mesh Network-on-Chip for Multicore Architectures," *IEEE Transactions on Embedded Computing Systems (TECS)*, 2013.
- [10] Q. Xu, S. Manipatruni, B. Schmidt, J. Shakya, and M. Lipson, "12.5 Gbit/s carrier-injection-based silicon microring silicon modulators," *Optics Express*, vol. 15, no. 2:22, p. 430436, 2007.
- [11] D. Vantrease, R. Schreiber, M. Monchiero, M. McLaren, N. Jouppi, M. Fiorentino, A. Davis, N. Binkert, R. Beausoleil, and J. Ahn, "Corona: System implications of emerging nanophotonic technology," in *Proceedings of the International Symposium on Computer Architecture (ISCA)*, 2008.
- [12] Y. Pan, P. Kumar, J. Kim, G. Memik, Y. Zhang, and A. Choudhary, "Firefly: Illuminating future network-on-chip with nanophotonics," in *Proceedings of the International Symposium on Computer Architecture (ISCA)*, 2009.
- [13] Y. Pan, J. Kim, and G. Memik, "Flexishare: Channel sharing for an energy efficient nanophotonic crossbar," in *Proceedings of the International Symposium on High Performance Computer Architecture (HPCA)*, 2010.

- [14] C. Chen and A. Joshi, "Runtime management of laser power in silicon-photonics multibus NoC," *IEEE Journal for Selected Topics Quantum Electronics*, vol. 19, no. 2, 2013.
- [15] Y. Vlasov, W. Green, and F. Xia, "High-throughput silicon nanophotonic wavelength-insensitive switch for on-chip optical networks," in *Proceedings of Nature Photonics*, vol. 2, no. 4, 2008.
- [16] G. Chen, H. Chen, M. Haurylau, N. Nelson, D. Albonese, M. Philippe, P. Fauchet, E. Friedman, and G. Eby, "Predictions of CMOS compatible on-chip optical interconnect," in *Proceedings of international workshop on System level interconnect prediction (SLIP)*, New York, NY, USA, 13-20, June 2005.
- [17] Q. Xu, B. Schmidt, S. Pradhan, and M. Lipson, "Micrometre-scale silicon electro-optic modulator," *Nature Letters*, vol. 435, 2005.
- [18] J. Ahn, M. Fiorentino, R. G. Beausoleil, N. Binkert, A. Davis, D. Fattal, N. P. Jouppi, M. McLaren, C. M. Santori, R. S. Schreiber, S. M. Spillane, D. Vantrease, and Q. Xu, "Devices and architectures for photonic chip-scale integration," *Applied Physics Materials Science & Processing*, vol. 95, no. 4, p. 989–997, 2009.
- [19] C. Nitta, M. Farrens, and V. Akella, "Addressing system-level trimming issues in on-chip nanophotonic networks," in *Proceedings of High Performance Computer Architecture (HPCA)*, 2011.
- [20] T. Pimpalkhute and S. Pasricha, "An application-aware heterogeneous prioritization framework for NoC based chip multiprocessors," in *Proceedings of International Symposium on Quality Electronic Design (ISQED)*, 2014.

- [21] C. Batten, A. Joshi, J. Orcutt, A. Khilo, and B. Moss, "Building Manycore Processor-to-DRAM Networks with Monolithic CMOS Silicon Photonics," in *Proceedings of High Performance Interconnects*, 2008.
- [22] S. K. Selvaraja, "Wafer-Scale Fabrication Technology for Silicon Photonic Integrated Circuits," *PhD thesis, Ghent University*, 2011.
- [23] Z. Li, M. Mohamed, X. Chen, E. Dudley, K. Meng, L. Shang, A. R. Mickelson, R. Joseph, M. Vachharajan, B. Schwartz, and Y. Sun, "Reliability Modeling and Management of Nanophotonic On-Chip Networks," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 20, no. 1, pp. 98-111, 2012.
- [24] P. Dong, S. Liao, D. Feng, H. Liang, D. Zheng, R. Shafiq, C. Kung, W. Qian, G. Li, X. Zheng, A. V. Krishnamoorthy, and M. Asghari, "Low V_{pp}, ultralow-energy, compact, high-speed silicon electro-optic modulator," *Optics Express*, vol. 17, no. 25, pp. 22484-22490, 2009.
- [25] S. V. R. Chittamuru, I. Thakkar, and S. Pasricha, "SWIFTNoC: A Reconfigurable Silicon-Photonic Network with Multicast Enabled Channel Sharing for Multicore Architectures," *ACM JETC*, vol. 13, no. 58, 2017.
- [26] S. V. R. Chittamuru, S. Desai, and S. Pasricha, "A Reconfigurable Silicon-Photonic Network with Improved Channel Sharing for Multicore Architectures," in *Proceedings of ACM Great Lakes Symposium on VLSI (GLSVLSI)*, 2015.
- [27] S. V. R. Chittamuru, D. Dang, S. Pasricha, and R. Mahapatra, "BiGNoC: Accelerating Big Data Computing with Application-Specific Photonic Network-on-Chip Architectures," in *IEEE Transactions on Parallel and Distributed Systems*, 2018.

- [28] S. V. R. Chittamuru and S. Pasricha, "Crosstalk Mitigation for High-Radix and Low-Diameter Photonic NoC Architectures," *IEEE Design and Test*, vol. 32, no. 3, pp. 29-39, 2015.
- [29] S. V. R. Chittamuru and S. Pasricha, "Improving Crosstalk Resilience with Wavelength Spacing in Photonic Crossbar-based Network-on-Chip Architectures," in *Proceedings of IEEE Midwest Symposium on Circuits and Systems (MWSCAS)*, 2015.
- [30] S. V. R. Chittamuru, I. Thakkar, and S. Pasricha, "Process Variation Aware Crosstalk Mitigation for DWDM based Photonic NoC Architectures," in *IEEE International Symposium on Quality Electronic Design (ISQED)*, 2016.
- [31] S. V. R. Chittamuru, I. Thakkar, and S. Pasricha, "PICO: Mitigating Heterodyne Crosstalk Due to Process Variations and Intermodulation Effects in Photonic NoCs," in *Proceedings of IEEE/ACM Design Automation Conference (DAC)*, 2016.
- [32] S. V. R. Chittamuru, I. Thakkar, and S. Pasricha, "HYDRA: Heterodyne Crosstalk Mitigation with Double Microring Resonators and Data Encoding for Photonic NoCs," in *TVLSI*, vol. 26, no. 1, 2018.
- [33] S. V. R. Chittamuru and S. Pasricha, "SPECTRA: A Framework for Thermal Reliability Management in Silicon-Photonic Networks-on-Chip," in *Proceedings of IEEE International Conference on VLSI Design (VLSID)*, 2016.
- [34] S. V. R. Chittamuru, D. Dang, R. Mahapatra, and S. Pasricha, "Islands of Heaters: A Novel Thermal Management Framework for Photonic NoCs," in *Proceedings of IEEE/ACM Asia and South Pacific Design Automation Conference (ASPDAC)*, 2017.

- [35] S. V. R. Chittamuru, I. Thakkar, and S. Pasricha, "LIBRA: Thermal and Process Variation Aware Reliability Management in Photonic Networks-on-Chip," in *Transactions on Multi-Scale Computing Systems*, 2018.
- [36] S. V. R. Chittamuru, I. Thakkar, and S. Pasricha, "Analyzing Voltage Bias and Temperature Induced Aging Effects in Photonic Interconnects for Manycore Computing," in *Proceedings of SLIP*, 2017.
- [37] S. V. R. Chittamuru, I. Thakkar, V. Bhatt, and S. Pasricha, "SOTERIA: Exploiting Process Variations to Enhance Hardware Security with Photonic NoC Architectures," in *DAC (under review)*, 2018.
- [38] I. Thakkar, S. V. R. Chittamuru, and S. Pasricha, "A Comparative Analysis of Front-End and Back-End Compatible Silicon Photonic On-Chip Interconnects," in *Proceedings of IEEE/ACM System-Level Interconnect Prediction (SLIP) Workshop*, 2016.
- [39] I. Thakkar, S. V. R. Chittamuru, and S. Pasricha, "Run-Time Laser Power Management in Photonic NoCs with On-Chip Semiconductor Optical Amplifiers," in *Proceedings of IEEE/ACM Network-on-Chips (NOCS)*, 2016.
- [40] I. Thakkar, S. V. R. Chittamuru, and S. Pasricha, "Mitigation of Homodyne Crosstalk Noise in Silicon Photonic NoC Architectures with Tunable Decoupling," in *Proceedings of ACM/IEEE CODES+ISSS*, 2016.
- [41] I. Thakkar, S. V. R. Chittamuru, and S. Pasricha, "Improving the Reliability and Energy-Efficiency of HighBandwidth Photonic NoC Architectures with Multilevel Signaling," in *Proceedings of NOCS*, 2017.

- [42] Y. Xie, M. Nikdast, J. Xu, W. Zhang, Q. Li, X. Wu, Y. Ye, X. Wang, and W. Liu, "Crosstalk Noise and Bit Error Rate Analysis for Optical Network-on-Chip," in *Proceedings of Design Automation Conference (DAC)*, 2010.
- [43] C. Bienia, S. Kumar, J. P. Singh, and K. Li, "The PARSEC Benchmark Suit: Characterization and," in *Proceedings of the international conference on Parallel architectures and compilation techniques (PACT)*, 2008.
- [44] Work Group, "International Technology Roadmap for Semiconductors," 2013.
- [45] F. Kreup, A. Graham, M. Liebau, G. Duesberg, R. Seidel, and E. Unger, "Carbon nanotubes for interconnect applications," in *Proceedings of the IEEE International Electron Devices Meeting (IEDM'04)*, 2004.
- [46] N. Srivastava and V. Banerjee, "Performance analysis of carbon nanotube interconnects for VLSI applications," in *Proceedings of the IEEE/ACM International Conference on Computer-Aided Design (ICCAD'05)*, 2005.
- [47] S. Pasricha, F. Kurdahi, and N. Dutt, "System level performance analysis of carbon nanotube global interconnects for emerging chip multiprocessors," in *Proceedings of the IEEE International Symposium on Nanoscale Architectures (NANOARCH'08)*, 2008.
- [48] B. De Vivo, P. Lamberti, G. Spinelli, and V. Tucci, "Reliable bounds for the propagation delay in VLSI nano interconnects based on Multi Wall Carbon Nano Tubes," in *Signal Propagation on Interconnects (SPI)*, 2010.
- [49] D. Zhao and Y. Wang, "SD-MAC: Design and synthesis of a hardware-efficient collision-free QOS-aware mac protocol for wireless network-on-chip," *IEEE Transactions on Computers*, vol. 57, no. 9, pp. 1230-1245, 2008.

- [50] S. Deb, K. Chang, A. Ganguly, Y. Xinmin, C. Teuscher, P. Pande, D. Heo, and B. Belzer, "Design of an efficient NoC architecture using millimeter-wave wireless links," in *Quality Electronic Design (ISQED)*, 2012.
- [51] P. Wettin, P.P. Pande, H. Deukhyoun, B. Belzer, S. Deb, and A. Ganguly, "Design space exploration for reliable mm-wave wireless NoC architectures," in *Application-Specific Systems, Architectures and Processors (ASAP)*, 2013.
- [52] H. Matsutani, M. Koibuchi, I. Fujiwara, T. Kagami, Y. Take, T. Kuroda, P. Bogdan, R. Marculescu, and H. Amano, "Low-latency wireless 3D NoCs via randomized shortcut chips," in *Proceedings of the conference on Design, Automation & Test in Europe (DATE '14)*, 2014.
- [53] P. P. Pande, A. Nojeh, and A. Ivanov, "T1B: Wireless NoC as interconnection backbone for multicore chips: Promises and challenges," in *System-on-Chip Conference (SOCC)*, 2014.
- [54] S. Bahirat and S. Pasricha, "A Particle Swarm Optimization Approach for Synthesizing Application-specific Hybrid Photonic Networks-on-Chip," in *IEEE International Symposium on Quality Electronic Design (ISQED'12)*, 2012.
- [55] J. Goodman, F. Leonberger, K. Sun-Yuan, and R. Athale, "Optical interconnects for vlsi systems," in *Proceedings of the IEEE*, vol. 72, no. 7, pp. 850-866, 1984.
- [56] D. Chiarulli, S. Levitan, R. Melhem, M. Bidnurkar, R. Ditmore, G. Gravenstreter, Z. Guo, J. Qao, and C. Teza, "Optoelectronic buses for high performance computing," in *Proceedings of IEEE*, vol. 82, no. 11, pp. 1701-1710, 1994.

- [57] J. Ha and T. Pinkston, "Speed demon: Cache coherence on an optical multichannel interconnect architecture," in *Journal for Parallel Distributed Computing*, vol. 41, no. 1, pp. 78-91, 1997.
- [58] E. Carrera and R. Bianchini, "OPTNET: A cost effective optical network for multiprocessors," in *Proceedings of International Conference on Supercomputing (ICS'98)*, New York, 1998.
- [59] A. Kodi and A. Louri, "RAPID: Reconfigurable and scalable All-Photonic Interconnect for Distributed shared memory multiprocessors," in *Journal for the Light-Wave Technology*, vol. 22, no. 9, pp. 2101-2110, 2004.
- [60] A. Joshi, C. Batten, Y-J. Kwon, S. Beamer, I. Shamim, K. Asanovic, and V. Stojanovic, "Silicon-photonic clos networks for global on-chip communication," in *Proceedings of ACM/IEEE International Symposium on Networks-on-Chip (NOCS)*, New York, 2009.
- [61] J. Psota, J. Eastep, J. Miller, T. Konstantakopoulos, M. Watts, M. Beals, J. Michel, K. Kimerling, and A. Agarwal, "ATAC: On-Chip Optical Networks for Multicore Processors," in *Proceedings of Boston Area Architecture Workshop*, 2007.
- [62] R. Morris and A. K. Kodi, "Power-efficient and high-performance multilevel hybrid nanophotonic interconnect for multicores," in *Proceedings of ACM/IEEE International Symposium on Networks-on-Chip (NOCS)*, 2010.
- [63] N. Kirman and J. F. Martinez, "A power-efficient all-optical on-chip interconnect using wavelength based oblivious routing," in *Proceedings of Architectural support for programming languages and operating systems (ASPLOS)*, 2010.

- [64] H. Li, S. Le Beux, G. Nicolescu, J. Trajkovic, and I. O'Connor, "Optical Crossbars on Chip, a comparative study based on worst-case propagation losses," *Concurrency and Computation: Practice and Experience*, vol. 26, no. 15, p. 2492–2503, 2014.
- [65] Y. Xue and P. Bogdan, "User Cooperation Network Coding Approach for NoC Performance Improvement," in *Proceedings of the 9th International Symposium on Networks-on-Chip (NOCS '15)*, 2015.
- [66] Z. Li, A. Qouneh, M. Joshi, W. Zhang, X. Fu, and T. Li, "Aurora: A Cross-Layer Solution for Thermally Resilient Photonic Network-on-Chip," in *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 23, no. 1, pp. 170-183, 2015.
- [67] D. Vantrease, N. Binkert, R. Schreiber, and M. H. Lipasti, "Light speed arbitration and flow control for nanophotonic interconnects," in *Proceedings of IEEE/ACM International Symposium on Microarchitecture (MICRO)*, 2009.
- [68] F. Cabarcas, A. Rico, Y. Etsion, and A. Ramirez, "Interleaving Granularity on High Bandwidth Memory Architecture for CMPs," in *International Conference on Embedded Computer Systems (SAMOS' 2010)*, 2010.
- [69] J. Kim, D. Park, T. Theocharides, N. Vijaykrishnan, and C. R. Das, "A low latency router supporting adaptivity for on-chip interconnects," in *Proceedings of annual conference on Design Automation (DAC'05)*, 2005.
- [70] N. E. Jerger, L. S. Peh, and M. H. Lipasti, "Virtual Circuit Tree Multicasting: A Case for On-Chip Hardware Multicast Support," in *Proceedings of ISCA*, 2008.
- [71] C. Li, M. Browning, P. V. Gratz, and S. Palermo, "Energy-efficient optical broadcast for nanophotonic networks-on-chip," in *Optical Interconnects Conference*, 2012.

- [72] N. Binkert , B. Beckmann , G. Black , S. K. Reinhardt , A. Saidi , A. Basu , J. Hestness , D. R. Hower , T.Krishna , S. Sadashti , R. Sen , K. Sewell , M. Shoaib , N. Vaish, M. D. Hill, and D. A. Wood, "The gem5 simulator," *ACM SIGARCH Computer Architecture News*, vol. 39, no. 2, pp. 1-7, 2011.
- [73] X. Zheng, D. Patil, J. Lexau, F. Liu, G. Li, H. Thacker, Y. Luo, I. Shubin, J. Li, J. Yao, P. Dong, D. Feng, M. Asghari, T. Pinguet, A. Mekis, P. Amberg, M. Dayringer, H. F. Moghadam, E. Alon, K. Raj, R. Ho, J. E. Cunningham, and A. V. Krishnamoorthy, "Ultra-efficient 10Gb/s hybrid integrated silicon photonic transmitter and receiver," *Optics Express*, vol. 19, no. 6, pp. 5172-5186, 2011.
- [74] P. Grani and S. Bartolini, "Design Options for Optical Ring Interconnect in Future Client Devices," *ACM Journal on Emerging Technologies in Computing Systems (JETC)*, vol. 10, no. 4, 2014.
- [75] C. Sun, C. O. Chen, G. Kurian, L. Wei, J. Miller, A. Agarwal, L. Peh, and V. Stojanovic, "DSENT - a tool connecting emerging photonics with electronics for opto-electronic networks-on-chip modeling," in *Proceedings of the ACM/IEEE International Symposium on Networks-on-Chip (NOCS'12)*, 2012.
- [76] S. Ghemawat and J. Dean, "MapReduce: Simplified Data Processing," *Comm. of the ACM*, vol. 51, no. 1, pp. 107-113, 2008.
- [77] Hadoop, "Hadoop," 2017. [Online]. Available: <https://hadoop.apache.org>.. [Accessed 2017].

- [78] M. Zaharia, M. Chowdhury, M. J. Franklin, S. Shenker, and I. Stoica, "Spark: Cluster computing with working sets," in *Proceedings of USENIX conference on Hot topics in cloud computing*, 2010.
- [79] J. Hamilton, "Cooperative expendable micro-slice servers: low cost low power servers for internet-scale services," in *CIDR*, 2009.
- [80] Y. Xia, T. S. E. Ng, and X. S. Sun, "Blast: Accelerating high-performance data analytics applications by optical multicast," in *IEEE Conference on Computer Communications (INFOCOM)*, 2015.
- [81] Breast Cancer Prediction and Prognosis, "Breast Cancer Prediction and Prognosis," 2017. [Online]. Available: https://www3.nd.edu/~steve/computing_with_data/17_Refining_kNN/refining_knn.html. [Accessed 2017].
- [82] M. Chowdhury, M. Zaharia, J. Ma, M. I. Jordan, and I. Stoica, "Managing Data Transfers in Computer Clusters with Orchestra," in *Proceedings of the ACM SIGCOMM*, 2011.
- [83] Text Mining, "Text Mining," [Online]. Available: <http://www.cs.umb.edu/~smimarog/textmining/datasets/>.
- [84] R. S. Tsay, "Analysis of Financial Time Series," John Wiley & Sons Inc., 2002.
- [85] Airline Data Set, "Airline Data Set," 2017. [Online]. Available: <http://www.stat.purdue.edu/~sguha/rhipe/doc/html/airline.html>. [Accessed 2017].
- [86] Gray Sort, "Gray Sort," 2017. [Online]. Available: <http://sortbenchmark.org/>. [Accessed 2017].

- [87] E. Fusella, J. Flich, and A. Cilardo, "Path Setup for Hybrid NoC Architectures Exploiting Flooding and Standby," *IEEE Transactions on Parallel and Distributed Systems*, vol. 28, no. 5, 2017.
- [88] A. Kulkarni, T. Abtahi, E. Smith, and T. Mohsenin, "Low energy sketching engines on many-core platform for big data acceleration," in *International Great Lakes Symposium on VLSI (GLSVLSI)*, 2016.
- [89] K. Kanoun, M. Ruggiero, D. Atienza, and M. Schaar, "Low Power and Scalable Many-Core Architecture for Big-Data Stream Computing," in *IEEE Computer Society Annual Symposium on VLSI (ISVLSI)*, 2014.
- [90] E. Painkras, L. A. Plana, J. Garside, S. Temple, F. Galluppi, C. Patterson, D. R. Lester, A. D. Brown, and S. B. Furber, "SpiNNaker: A 1-W 18-core system-on-chip for massively-parallel neural network simulation," in *IEEE Journal of Solid-State Circuits*, vol. 48, no. 8, pp. 1943-1953, 2013.
- [91] S. Carrillo, J. Harkin, L. J. McDaid, F. Morgan, S. Pande, S. Cawley, and B. McGinley, "Scalable hierarchical network-on-chip architecture for spiking neural network hardware implementations," in *IEEE Transactions on Parallel and Distributed Systems*, vol. 24, no. 22, 2013.
- [92] V. Dmitri and R. Ginosar, "Network-on-chip architectures for neural networks," in *Proceedings of the ACM/IEEE International Symposium on Networks-on-Chip (NOCS)*, 2010.
- [93] A. Firuzan, M. Modarressi, and M. Daneshtalab, "Reconfigurable communication fabric for efficient implementation of neural networks," in *Proceedings of IEEE ReCoSoC*, 2015.

- [94] J. Lee, S. Li, H. Kim, and S. Yalamanchili, "Design Space Exploration of On-chip Ring Interconnection for a CPU-GPU Heterogeneous Architecture," in *Journal of Parallel and Distributed Computing*, vol. 73, no. 12, 2013.
- [95] W. Choi, K. Duraisamy, R. G. Kim, J. R. Doppa, P. P. Pande, R. Marculescu, and D. Marculescu, "Hybrid Network-on-Chip Architectures for Accelerating Deep Learning Kernels on Heterogeneous Manycore Platforms," in *Proc. of CASES*, 2016.
- [96] Z. Wang, S. Zhang, B. He, and W. Zhang, "Melia: A Mapreduce Framework on OpenCL-Based FPGAs," in *IEEE Trans. Parallel and Distri. Sys (TPDS)*, vol. 27, no. 12, 2016.
- [97] J. Dongarra, "Report on the Sunway Taihulight System," in *Technical Report UT-EECS-16-742*, 2016.
- [98] T. Krishna, L. Peh, B. M. Beckmann, S. K. Reinhardt, "Towards the Ideal On-chip Fabric for 1-to-Many and Many-to-1 Communication," in *Proceedings of IEEE/ACM International Symposium on Microarchitecture (MICRO)*, 2011.
- [99] Amazon EC2, "Amazon EC2," 2017. [Online]. Available: <http://aws.amazon.com/ec2>. [Accessed 2017].
- [100] L.H.K. Duong, M. Nikdast, S. Le Beux, J. Xu, X. Wu, Z. Wang, and P. Yang, "A Case Study of Signal-to-Noise Ratio in Ring-Based Optical Networks-on-Chip," in *IEEE Design and Test*, vol. 31, no. 5, 2014.
- [101] C. Chen, "Waveguide crossings by use of multimode tapered structures," in *Proceedings of Wireless and Optical Communications Conference (WOCC)*, 2012.
- [102] Q. Xu, B. Schmidt, J. Shakya, and M. Lipson, "Cascaded silicon micro-ring modulators for wdm optical interconnection," in *Optics Express*, vol. 14, no. 20, pp. 9431-9436, 2006.

- [103] M. Nikdast, J. Xu, L. H. K. Duong, X. Wu, Z. Wang, X. Wang, and Z. Wang, "Fat-Tree-Based Optical Interconnection Networks Under Crosstalk Noise Constraint," in *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 23, no. 1, 2014.
- [104] S. Xiao, M. H. Khan, H. Shen, and M. Qi, "Modeling and measurement of losses in silicon-on-insulator resonators and bends," in *Optics Express*, vol. 15, no. 17, pp. 10553-10561, 2007.
- [105] Y. Xu, J. Yang, and R. Melhem, "Tolerating process variations in nanophotonic on-chip networks," in *International Symposium on Computer Architecture (ISCA)*, 2012.
- [106] P. K. Kaliraj, P. Sieber, A. Ganguly, I. Datta, and D. Datta, "Performance evaluation of reliability aware photonic Network-on-Chip architectures," in *Proceedings of International Green Computing Conference (IGCC)*, 2012.
- [107] I. Datta, D. Datta, and P. P. Pande, "Design Methodology for Optical Interconnect Topologies in NoCs With BER and Transmit Power Constraints," in *Journal of Lightwave Technology*, vol. 32, no. 1, 2014.
- [108] K. Padmaraju, X. Zhu, L. Chen, M. Lipson and K. Bergman, "Intermodulation Crosstalk Characteristics of WDM Silicon Microring Modulators," in *IEEE Photonics Technology Letters*, vol. 26, no. 14, pp. 1478 - 1481, 2014.
- [109] Y. Xu and S. Pasricha, "Silicon nanophotonics for future multicore architectures: opportunities and challenges," in *IEEE Design and Test*, 2014.
- [110] K. Preston, N. Sherwood-Droz, J. S. Levy, and M. Lipson, "Performance guidelines for WDM interconnects based on silicon microring resonators," in *Conference on Lasers and Electro-Optics (CLEO)*, 2011.

- [111] R. G. Beausoleil, "Large-Scale Integrated Photonics for High-Performance Interconnects," *in ACM JETC*, vol. 7, no. 2, 2011.
- [112] S. R. Sarangi, B. Greskamp, R. Teodorescu, J. Nakano, A. Tiwari, J. Torrellas, "VARIUS: A Model of Process Variation and Resulting Timing Errors for Microarchitects," *in IEEE Transactions on Semiconductor Manufacturing*, vol. 21, no. 1, pp. 3-13, 2008.
- [113] K. Padmaraju and K. Bergman, "Resolving the thermal challenges for silicon microring resonator devices," *in Nanophotonics*, vol. 2, no. 4, 2013.
- [114] CACTI 6.5, "CACTI 6.5," [Online]. Available: <http://www.hpl.hp.com/research/cacti/>.
- [115] L. H. K. Duong, M. Nikdast, J. Xu, Z. Wang, Y. Thonnart, S. Le Beux, P. Yang, X. Wu, and Z. Wang, "Coherent Crosstalk Noise Analyses in Ringbased Optical Interconnects," *in Design, Automation & Test in Europe Conference & Exhibition (DATE)*, 2015.
- [116] M. Bahadori, D. Nikolova, Sébastien Rumley, C. P. Chen and K. Bergman, "Optimization of Microring-based Filters for Dense WDM Silicon Photonic Interconnects," *in IEEE Optical Interconnects Conference (OI)*, 2015.
- [117] B. G. Lee, B. A. Small, K. Bergman, Q. Xu, and M. Lipson, "Transmission of high-data-rate optical signals through a micrometer-scale silicon ring resonator," *in Optics Letters*, vol. 31, no. 18, pp. 2701-2703, 2006.
- [118] C. Sun, et al., "Single-chip microprocessor that communicates directly using light," *in Nature*, vol. 528, pp. 24-31, 2015.
- [119] P. K. Tien, "Light waves in thin films and integrated optics," *in Applied Optics*, vol. 10, no. 11, 1971.

- [120] R. Hendry, D. Nikolova, S. Rumley, N. Ophir, and K. Bergman, "Physical Layer Analysis and Modeling of Silicon Photonic WDM Bus Architectures," in *HiPEAC Workshop*, 2014.
- [121] W. Bogaerts et al., "Silicon microring resonators," *Laser and Photonics Reviews*, vol. 6, no. 1, 2012.
- [122] Lumerical Solutions Inc., "Lumerical Solutions Inc.," [Online]. Available: [http://www.lumerical.com/tcad-products/mode/..](http://www.lumerical.com/tcad-products/mode/)
- [123] M. Bahadori et al., "Optimization of Microring-based Filters for Dense WDM Silicon Photonic Interconnects," in *IEEE OIC*, 2015.
- [124] J. E. Heebner, "Nonlinear optical whispering gallery microresonators for photonics," *Ph.D. dissertation, Univ. Rochester, NY*, 2003.
- [125] D. A. Neamen, *Semiconductor Physics and Devices: Basic Principles*, McGraw-Hil, 2002.
- [126] G. T. Reed and A. P. Knights, , *Silicon Photonics: An Introduction*, Wiley, 2004.
- [127] C. Li et al., "Silicon Photonic Transceiver Circuits With Microring Resonator Bias-Based Wavelength Stabilization in 65 nm CMOS," in *JSSC*, vol. 49, no. 6, 2014.
- [128] N. Ophir, C. Mineo, D. Mountain, and K. Bergman, "Silicon photonic microring links for high-bandwidth-density, low-power chip I/O," in *IEEE Micro*, vol. 33, no. 1, 2013.
- [129] D. G. Rabus , *Integrated Ring Resonators: The Compendium*, Springer, 2007.
- [130] A. Sridhar, A. Vincenzi, M. Ruggiero, T. Brunswiler, and D. Atienza, "3D-ICE: Fast compact transient thermal modeling for 3D ICs with inter-tier liquid cooling," in *Proceedings of IEEE/ACM International Conference on Computer-Aided Design (ICCAD)*, 2010.

- [131] S. C. Woo, M. Ohara, E. Torrie, J. P. Singh, and A. Gupta, "The SPLASH-2 programs: characterization and methodological considerations," in *Proceedings of International Symposium on Computer Architecture (ISCA)*, 1995.
- [132] S. S. Djordjevic, K. Shang, B. Guan, S. T. S. Cheung, L. Liao, J. Basak, H. Liu, and S. J. B. Yoo, "CMOS-compatible, athermal silicon ring modulators clad with titanium dioxide," in *Optics Express*, vol. 21, no. 12, pp. 13958-13968, 2013.
- [133] T. Zhang, J. L. Abellan, A. Joshi, and A. K. Coskun, "Thermal management of manycore systems with silicon-photonics networks," in *Proceedings of Design, Automation and Test in Europe Conference and Exhibition (DATE)*, 2014.
- [134] D. Dang et al., "PID-controlled Heater-based Thermal Management in Photonic Network-on-Chip," in *Proceedings of IEEE ICCD*, 2015.
- [135] PIDLAB, "PID LAB," 2016. [Online]. Available: <http://www.pidlab.com/en/>. [Accessed 2016].
- [136] D. Harris et al., "Support Vector Regression Machines," in *Advances in Neural Information Processing Systems (NIPS)*, MIT Press, 1996.
- [137] H. Drucker, C. J. C. Burges, L. Kaufman, A. J. Smola, and V. Vapnik, "Support Vector Regression Machines," in *Proceedings of Advances in Neural Information Processing Systems (NIPS)*, MIT Press, 1996.
- [138] B. E. Boser et al., "A training algorithm for optimal margin classifiers," in *5th Annual Workshop Comput. Learning Theory*, 1992.
- [139] I. Yeo, C. C. Liu, and E. J. Kim, "Predictive dynamic thermal management for multicore systems," in *Proceedings of Design Automation Conference (DAC)*, 2008.

- [140] IPKISS, "IPKISS - a generic and modular software framework for parametric design," 2015. [Online]. Available: <http://www.ipkiss.org/>. [Accessed 2016].
- [141] T. E. Carlson, W. Heirmant, and L. Eeckhout, "Sniper: Exploring the level of abstraction for scalable and accurate parallel multi-core simulation," in *Proceedings of International Conference for High Performance Computing, Networking, Storage and Analysis*, 2011.
- [142] S. Li, J. H. Ahn, R. D. Strong, J. B. Brockman, D. M. Tullsen, and N. P. Jouppi, "McPAT: An integrated power, area, and timing modeling framework for multicore and manycore architectures," in *Proceedings of IEEE/ACM International Symposium on Microarchitecture*, 2009.
- [143] Mt-Berlin, "Crystal quartz (SiO₂) and Fused Silica," 2014. [Online]. Available: http://www.mt-berlin.com/frames_cryst/descriptions/quartz.htm. [Accessed 2016].
- [144] A. Kumar, L. Shang, L. Peh, and N. K. Jha, "HybDTM: A Coordinated Hardware-Software Approach for Dynamic Thermal Management," in *Proceedings of Design Automation Conference (DAC)*, 2006.
- [145] J. L. Abellan et al., "Adaptive Tuning of Photonic Devices in a Photonic NoC Through Dynamic Workload Allocation," in *IEEE TCAD*, vol. 1, no. 99, 2016.
- [146] A. K. Coskun, T. Rosing, K. A. Whisnant, and K. C. Gross, "Static and Dynamic Temperature-Aware Scheduling for Multiprocessor SoCs," in *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 16, no. 9, pp. 1127 - 1140, 2008.
- [147] H. Hanson et al., "Thermal response to DVFS: analysis with an Intel Pentium M," in *Proceedings of ISLPED*, 2007.

- [148] V. Hanumaiah, S. Vrudhula, and K. S. Chatha, "Maximizing performance of thermally constrained multi-core processors by dynamic voltage and frequency control," in *Proceedings of ICCAD*, 2009.
- [149] S. Herbert and D. Marculescu, "Analysis of dynamic voltage/frequency scaling in chip-multiprocessors," in *ISLPED*, 2007.
- [150] K. K. Rangan, G. Y. Wei, and D. Brook, "Thread motion: fine-grained power management for multi-core systems," in *ISCA*, 2009.
- [151] E. Ipek et al., "Core fusion: accommodating software diversity in chip multiprocessors," in *proceedings of ISCA*, 2009.
- [152] A. Fourmigue, et al., "Efficient transient thermal simulation of 3D ICs with liquid-cooling and through silicon vias," in *DATE*, 2014.
- [153] M. M. Sabry, A. Sridhar, and D. Atienza, "Thermal balancing of liquidcooled 3D-MPSoCs using channel modulation," in *DATE*, 2012.
- [154] A. K. Coskun et al., "Energy-efficient variable-flow liquid cooling in 3D stacked architectures," in *DATE*, 2010.
- [155] A. K. Coskun, J. L. Ayala, D. Atienza, and T. S. Rosing, "Modeling and dynamic management of 3D multicore systems with liquid cooling," in *Proceedings of International Conference on Very Large Scale Integration (VLSI-SoC)*, 2009.
- [156] B. Raghunathan et al., "Cherry-picking: Exploiting process variations in dark-silicon homogeneous chip multi-processors," in *DATE*, 2013.
- [157] N. Kapadia and S. Pasricha, "VARSHA: Variation and reliability-aware application scheduling with adaptive parallelism in the dark-silicon era," in *DATE*, 2015.

- [158] B. Guha, et al., "Cmos-compatible athermal silicon microring resonators," in *Optic Express*, vol. 18, no. 4, 2010.
- [159] C. T. DeRose, M. R. Watts, D. C. Trotter, D. L. Luck, G. N. Nielson, and R. W. Young, "Silicon microring modulator with integrated heater and temperature sensor for thermal control," in *Proceedings of Lasers and Electro-Optics (CLEO) and Quantum Electronics and Laser Science Conference (QELS)*, 2010.
- [160] M. Georgas et al., "Addressing link-level design tradeoffs for integrated photonic interconnects," in *CICC*, 2011.
- [161] M. V. Beigi and G. Memik, "Therma: Thermal-aware Run-time Thread Migration for Nanophotonic Interconnects," in *ISLPED*, 2016.
- [162] M. Mohamed, Z. Li, X. Chen, L. Shang, and A. R. Mickelson, "Reliability-Aware Design Flow for Silicon Photonics On-Chip Interconnect," in *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 22, no. 8, pp. 1763 - 1776, 2014.
- [163] M. Mohamed et al., "Power-Efficient Variation-Aware Photonic OnChip Network Management," in *ISLPED*, 2010.
- [164] R. Wu et al., "Variation-Aware Adaptive Tuning for Nanophotonic Interconnects," in *proceedings of ICCAD*, 2015.
- [165] P. P. Absil et al., "Silicon photonics integrated circuits: a manufacturing platform for high density, low power optical i/os," in *Optics Express*, vol. 23, no. 7, 2015.
- [166] M. Nikdast, et al., "Modeling fabrication non-uniformity in chip-scale silicon photonic interconnects," in *DATE*, 2016.

- [167] M. A. Alam, et al., "A comprehensive model for PMOS NBTI degradation," in *Microelectronics Reliability*, vol. 45, 2005.
- [168] H. Kufluoglu, "MOSFET degradation due to negative bias temperature instability (NBTI) and hot carrier injection (HCI) and its implications for reliability-aware VLSI design," PhD thesis, Purdue University, 2007.
- [169] H. Kufluoglu, et al., "Theory of interface-trap-induced NBTI degradation for reduced cross section MOSFETs," in *IEEE TED*, 2006.
- [170] S. Bahirat and S. Pasricha, "Exploring Hybrid Photonic Networks-on-Chip for Emerging Chip," in *IEEE/ACM International Conference on Hardware/Software Codesign and System Synthesis (CODES+ISSS'09)*, 2009.
- [171] S. Ogawa et al., "Generalized diffusion-reaction model for the low-field charge build up instability at the Si-SiO₂ interface," in *Phys. Review B*, vol. 51, no. 7, 1995.
- [172] M. Lipson , "Guiding, Modulating, and Emitting Light on Silicon—Challenges and Opportunities," in *Journal for Light Wave Technology*, vol. 23, no. 12, 2005.
- [173] M. A. Alam, et al., "A comprehensive model for PMOS NBTI degradation: Recent progress," in *Microelectronics Reliability*, vol. 47, 2007.
- [174] M. Cho et al., "Power Multiplexing for Thermal Field Management in Many-Core Processors," in *IEEE Transactions on Components, Packaging and Manufacturing Technology*, vol. 3, no. 1, 2013.
- [175] R. Chakraborty, S. Narasimhan, S. Bhunia, "Hardware Trojan: Threats and emerging solutions," in *Proc. HLDVT*, 2009.

- [176] M. Tehranipoor and F. Koushanfar, "A Survey of Hardware Trojan Taxonomy and Detection," in *IEEE Design & Test*, 2009.
- [177] S. Skorobogatov and C. Woods, "Breakthrough silicon scanning discovers backdoor in military chip," in *Proceedings of CHES*, 2012.
- [178] L. Benini and G. De Micheli, "Networks on chip: A new paradigm for systems on chip design," in *Proceedings of Design Automation and Test in Europe (DATE)*, 2002.
- [179] W. J. Dally and B. Towles, "Route packets, not wires," in *Proceedings of Design Automation conference (DAC'01)*, Las Vegas, 2001.
- [180] D. Miller, "Rationale and challenges for optical interconnects to electronic chips," in *Proceedings of IEEE*, vol. 88, no. 6, pp. 728-749, 2002.
- [181] J. D. Owens, W. J. Dally, R. Ho, D. N. Jayasimha, S. W. Keckler, and L. Peh, "Research challenges for on-chip interconnection networks," in *IEEE Micro*, vol. 27, no. 5, pp. 96-108, 2007.
- [182] Y. A. Vlasov, "Silicon CMOS-integrated nano-photonics for computer and data communications beyond 100G," in *IEEE Communications Magazine*, vol. 50, no. 2, 2012.
- [183] D. Dang et al., "Mode-division-Multiplexed Photonic Router for High Performance NoC," in *Proceedings of IEEE VLSI*, 2015.
- [184] M. Petracca, B. G. Lee, K. Bergman, and L. P. Carloni, "Design exploration of optical interconnection networks for chip multiprocessors," in *Proceedings of IEEE Symposium on High Performance Interconnects (HOTI)*, 2008.

- [185] M. J. Cianchetti, J. C. Kerekes, and D. H. Albonesi, "Phastlane: a rapid transit optical routing network," in *Proceedings of International Symposium on Computer Architecture (ISCA)*, 2009.
- [186] S. Pasricha and S. Bahirat, "OPAL: A multi-layer hybrid photonic NoC for 3D ICs," in *Asia and South Pacific Design Automation Conference (ASP-DAC)*, 2011.
- [187] J. P. Diguët and S. Evain, "From NoC security analysis to design solutions," in *Proceedings of SiPS*, 2005.
- [188] D. M. Ancajas, et al., "Fort-NoCs: Mitigating the Threat of a Compromised NoC," in *Proceedings of DAC*, 2014.
- [189] T. Boraten, et al., "Mitigation of Denial of Service Attack with Hardware Trojans in NoC Architectures," in *Proceedings of IPDPS*, 2016.
- [190] C. H. Gebotys, et al., "A framework for security on NoC technologies," in *Proceedings of ISVLSI*, 2003.
- [191] H. K. Kapoor, et al., "A Security Framework for NoC Using Authenticated Encryption and Session Keys," in *CSSP*, 2013.
- [192] K. Padmaraju et al., "Wavelength Locking and Thermally Stabilizing Microring Resonators Using Dithering Signals," in *Journal for Light wave Technology*, vol. 32, no. 3, 2013.
- [193] X. Xue et al., "Microresonator Kerr frequency combs with high conversion efficiency," in *Laser & Photonics Rev*, vol. 11, no. 1, 2017.