

Carnegie Mellon University
Research Showcase @ CMU

Dissertations

Theses and Dissertations

12-2011

Theoretical Foundations of Multicore Systems Design: A Dynamical Systems Perspective

Paul Bogdan

Carnegie Mellon University, pbogdan@andrew.cmu.edu

Follow this and additional works at: <http://repository.cmu.edu/dissertations>



Part of the [Electrical and Computer Engineering Commons](#)

Recommended Citation

Bogdan, Paul, "Theoretical Foundations of Multicore Systems Design: A Dynamical Systems Perspective" (2011). *Dissertations*. Paper 183.

This Dissertation is brought to you for free and open access by the Theses and Dissertations at Research Showcase @ CMU. It has been accepted for inclusion in Dissertations by an authorized administrator of Research Showcase @ CMU. For more information, please contact research-showcase@andrew.cmu.edu.

Theoretical Foundations of Multicore Systems Design: A Dynamical Systems Perspective

Submitted in partial fulfillment of the requirements for
the degree of
Doctor of Philosophy
in
Electrical and Computer Engineering

Paul Bogdan

B.S., Automatic Control and Computer Science, “Politehnica” University of Bucharest

Carnegie Mellon University
Pittsburgh, PA

December, 2011

Acknowledgments

First and foremost, I would like to express my sincere gratitude to my thesis advisor and mentor Radu Marculescu for his perpetual motivation and support throughout my Ph.D. studies at Carnegie Mellon University. Thanks to his invaluable guidance, I have dared to bridge the gaps between academic disciplines and learned how to formulate advanced research questions. He has inspired me to think outside the box and dream of novel solutions to complex problems. The countless research discussions I have had with Radu Marculescu have significantly contributed to the fascinating correlations I have managed to discover between computer engineering, computer science, statistical physics, and applied mathematics. I am thankful to him for nurturing my intellectual curiosity and encouraging me to develop my own research path. Together with him, I have worked on the most fascinating problems with wide-range implications for biological, technological, and social systems.

Secondly, I consider myself fortunate to have received feedback on my research from my thesis committee members: Radu Balan, Shawn Blanton, Petru Eles, Peter Feldmann, Alan Frize, and Diana Marculescu. I am particularly grateful to Radu Balan (University of Maryland), who has shaped my analytical thinking in the early stages of my research endeavors. My long conversations with him during my internship at Siemens Corporate Research, Princeton have determined me to pursue doctoral studies. Radu Balan's challenging and captivating questions have haunted me for a long time after this first research stage and have represented a persistent beacon of inspiration throughout my doctoral period. I have also been fortunate to benefit from the advice of valuable faculty members of Carnegie Mellon University. Shawn Blanton's insightful feedback on my qualifying exam, as well as on my proposal and defense, has proven extremely valuable. I would also like to thank Diana Marculescu for her inspiring research feedback and wise suggestions for my PhD thesis. I am greatly indebted to Petru Eles (Linköping University) for taking the time to examine my research endeavors and offer me crucial suggestions for improving my approach and developing the applicability of fractional calculus to real-world problems. I am also thankful to Peter Feldman (IBM T.J. Watson) for his helpful input on optimal control theory and its application to VLSI design automation. Last, but not least, I am grateful to Alan Frieze (Carnegie Mellon University) for offering me insightful advice on random walks on graphs.

I have been privileged to benefit from the guidance of brilliant professors and researchers from Carnegie Mellon University and advanced centers of technological research. I am especially grateful to Rob Rutenbar and Don Thomas (Carnegie Mellon University) for their stimulating questions during the first years of my doctoral period when I was endeavoring to formulate my thesis topic. My conversations with Michael Kishinevsky (Intel) have also been extremely helpful and inspiring. I am thankful to him for sharing his technological perspective on my theoretical propositions and for developing my thinking on computer engineering problems. I would also like to express my sincere gratitude to Justinian Rosca (Siemens Corporate Research) for his crucial mentoring of my predoctoral research. Moreover, my advanced studies in Electrical and Computer Engineering would not have been possible without a strong foundation in mathematics and physics developed during

my undergraduate studies at Politehnica University of Bucharest and my high school studies at Andrei Saguna National College. I would like to thank all the professors who have contributed to my early academic formation: Florina Saon, Romeo Ilie, Emil Rabanca, Radu Chisleag, Cristian Oara, Radu Stefan, Corneliu Popeea, Boris Jora, Radu Gologan, Dorin Carstoiu, Bogdan Dumitrescu, Radu Dobrescu and Dumitru Popescu.

Above all, I have been extremely blessed to count on the unconditional love and support of my parents, Cristina and Nicolae, and my sister Corina. My family members have considerably helped me become a responsible individual and pursue challenging educational goals. My father has offered me training in algebra and logic from an early age and my mother helped me search for answers to inquisitive questions, as well as offered me explanations about the world, which introducing me to Euclidian geometry and physics concepts, such as Brownian motion.

My life as a graduate student at Carnegie Mellon University would have been much harder without the heartfelt support of many friends. I would like to extend my warm appreciation to the colleagues and friends who have made my Ph.D. life more fruitful and exciting. I thank Goksel Dedeoglu for all his support when I first arrived in Pittsburgh, as well as for his friendship and support. I am particularly thankful to my friend Alexandre Stauffer with whom I have enjoyed memorable road trips and insightful research discussions. I would also like to thank my current colleagues (Cory Bevilacqua, Radu David, Guopeng Wei), former System Level Design group members (Kaushik Chaubal, Po-Yuan Chen, Chen-Ling Chou, Shun-ping Chiu, Rafael Tornero Gavila, Ting-Chun Huang, Siddharth Jain, Jung-Chun Kao, Hyung Gyu Lee, Hiroki Matsutani, Umit Ogras, Anoop Ramakrishna, Nick Zamora), and Carnegie Mellon University fellows (Mudit Bhargava, Puru Choudhary, Siddharth Garg, Rachit Gupta, Da-Cheng Juan, Osei Poku, Phillip Stanley-Marbell, Emil Talpes, Kai-Chiang Wu, Soner Yaldiz), as well as my advisees (Siddharth Jain, Edward Ma, Guopeng Wei) whose enthusiasm for research and hard work have been highly rewarding. I am also thankful to Twan Basten for his insightful comments on complexity theory, embedded system design, and performance analysis. Last but not least, I am thankful to doctors Lisa, Michael, and Rishi for helping me recover my vision, as well as to all the CMU staff members (Adam, Elaine, Judy, Lori, Lynn, Lyz, Marian, Reenie, Susan, Tara) who have ensured that I have all the logistical and technological support necessary during my PhD studies.

I am very grateful to my girlfriend Cristina Albu for her love, encouragement, advice, and understanding throughout my doctoral studies. Without her, this period would have been infinitely harder. I would also like to thank all my friends in Pittsburgh (Corina Balut, Elena Canepa, Tudor Dumitras, Shivayogi Hiremath, Izabel and Joe Galliera, Ana-Maria Iosif, Diana Luca, Costin Tanase, Octavian Vasilescu, Madalina Veres) who made my experience away from home very enjoyable.

Finally, I would like to express my gratitude to the agencies who have contributed to the funding of my research, namely National Science Foundation, Gigascale Systems Research Focus Center, Semiconductor Research Corporation, and the Roberto Rocca Education Program.

Abstract

THEORETICAL FOUNDATIONS OF MULTICORE SYSTEMS: A DYNAMICAL SYSTEMS PERSPECTIVE

by

Paul Bogdan

Doctor of Philosophy in Electrical and Computer Engineering, Carnegie Mellon University
Professor Radu Marculescu, Chair

The proliferation of complex phenomena and the tightening competition for limited resources are two fundamental challenges for the modeling, analysis, and optimization of dynamical processes taking place in networked environments/architectures. Modes of collective and competitive behavior can be noticed across a wide array of social, biological, and technological contexts. From urban crowds to bacterial colonies, from brain neurons to human cells and even electron-hole interactions in semiconductors, dynamical phase transitions influence the macroscopic behavior of complex networks. To address these challenges, we focus on understanding, modeling, analyzing, and optimizing large-scale interconnected systems, such as future thousand-core Networks-on-Chip (NoC), biologically propelled Cyber-Physical Systems (CPS) consisting of micro-robotic swarms, or biological networks of stem cells, for performance, power, or fault-tolerance.

Enabled by recent advances in CMOS technology, the integration of tens and soon thousands of heterogeneous processing cores communicating via the NoC paradigm brings into the discussion the traffic modeling problem. Traffic modeling is of crucial importance for both static and dynamic NoC design and optimization problems such as topology selection and resource allocation, mapping, scheduling, routing or power management. The approaches proposed so far exhibit major limitations. For example, many of the queueing theory based modeling and optimization approaches proposed for NoC architectures ignore the traffic characteristics (*e.g.*, non-stationarity, fractality). In many situations, these models lead to buffer overflow or deadline missing situations and so poor (non-optimal) performance levels.

In this thesis, we show how this state of affairs can be changed by embracing a statistical physics inspired approach in order to insure accurate network traffic characterization. By using an analogy between a thermodynamic gas and a networked multicore architecture, our model captures the relevant traffic characteristics (*e.g.*, fractality, non-stationarity) via a dynamical master equation. Our approach not only leads to a more accurate estimation of performance metrics over Markovian models, but also helps at defining new state space model for dynamical systems that can be used for online optimization. Besides being an important contribution by itself, this radically new approach enhances the efficiency of resource allocation in nanoscale networks and overcomes the prior limitations of performance analysis approaches based on queuing models.

Building on statistical physics grounds, we model and analyze a biologically inspired communication protocol, namely the stochastic communication protocol. Under the stochastic communication protocol aiming at mitigating the nanoscale challenges (*e.g.*, particle hits, cross-talk) in multi-core

platforms, each node in the network disseminates packets multiple times via multiple paths. Hence, fault-tolerance is enforced at system-level by exploiting path diversity. To characterize such a dynamic behavior, we concentrate on estimating various performance metrics via a master equation approach capturing communication events such as packet duplication, packet transmission, and packet corruption. The proposed statistical physics model allows us to identify the benefits of this protocol for future communication fabrics.

To address the power consumption issues in large scale on-chip networks, we formulate the power and peak temperature management of heterogeneous NoC platforms as constrained finite horizon fractal optimal control problem. We show not only that fractal characteristics can be accounted for via fractional calculus state space models, but also that the online controller can be reduced to a linear program and efficiently computed via parallel algorithms. Our approach not only leads to significant power savings, but it also opens new avenues for dynamic optimization of large-scale systems exhibiting fractal dynamics.

Based on the proposed framework for modeling, analysis, and optimization of dynamical processes taking place on networked architectures, we formulate general guidelines for CPS design. As a concrete CPS example, we consider the design problem of the control algorithm of a pacemaker, which takes into account at run-time the fractal characteristics exhibited by heart rate variability.

In summary, this thesis offers a statistical physics view on using the network-paradigm in multi-core and cyber-physical system design. The results and discussion presented herein can be further extended to other classes of systems and applications. One research direction is represented by modeling and optimization of bacteria propelled micro-robotic swarms. Another research direction concerns the modeling of human dynamic processes such as car traffic, which can enable road structure optimization. Moreover, by relying on this statistical physics inspired framework, we can define models for biological communication and heterogeneous population growth with applications in regenerative medicine.

Contents

1	Introduction	1
1.1	Multicore Systems for Embedded and Cyber-Physical Applications	2
1.1.1	Network-on-Chip Architectures	3
1.1.2	Cyber-Physical Systems	4
1.2	Theoretical Foundations	5
1.2.1	Statistical Physics	5
1.2.2	Fractional Calculus	7
1.2.3	A Glimpse into Fractals	8
1.3	Thesis Overview and Research Objectives	10
1.4	Thesis Organization	12
2	A Dynamic Perspective on Networked-based Architectures Design	14
2.1	The Big and the Small: Technology Implications on NoC Design Flow	14
2.2	Application Modeling	17
2.3	Networks-on-Chip Architecture Modeling	19
2.4	Challenges of Designing Thousand Core Network-based Architectures	20
3	Multi-core Traffic Characterization	23
3.1	Rationale for Multi-core Traffic Monitoring	23
3.2	Related Work and Major Contributions	24
3.3	A Statistical Physics Inspired Model for NoC Traffic	27
4	Implications of Traffic Fractality	33
4.1	Analytical Implications of the Master Equation to Multi-core Traffic Modeling . . .	33
4.2	Critical Phenomena in Multi-core Systems	34
4.3	Implications of NoC Traffic Multi-fractality on Buffer Sizing	36
4.4	Implications of Traffic Multi-fractality on Estimating the Node-to-Node Latency . .	37
4.5	Where Does the Traffic Model Fit in the NoC Design and Optimization Flow? . . .	39
5	Modeling and Analysis of Fault-tolerant NoC Communication Protocols	41
5.1	Biologically Inspired Communication Protocols for NoCs: Stochastic Communication	41

5.2	Related Work and Major Contributions	42
5.3	A Branching and Annihilating Random Walks Approach to Modeling Fault-Tolerant Communication Protocols	44
5.4	Performance Analysis: Hitting Probabilities and Hitting Times	48
6	Dynamic Optimization Techniques for Networked-based Architectures	50
6.1	The Quest for Limiting Watts Expenditure at Nanoscale	50
6.2	Power Management for NoC Architectures under Highly Variable Workloads	52
6.3	Algorithmic Perspective and Experimental Results	55
6.4	Power and Peak Temperature Optimization of NoCs	61
6.5	Experimental results	71
7	Modeling, Analysis and Optimization of Cyber-Physical Systems	76
7.1	State of The Art in CPS Design and Optimization	76
7.2	Incorporating Time as Essential Component in Future Computing Platforms	78
7.3	Dynamic Optimization Methodologies for Cyber-Physical Systems	82
7.4	A Dynamic Optimization Case Study: Fractal Optimal Control for Bioimplantable Devices	83
8	Conclusion and Future Research Directions	91
8.1	Major Contributions of this Dissertation	91
8.2	Beyond Silicon: Future Research Directions	93
8.3	Modeling, Analysis and Optimization of Bacteria-propelled Micro-robotic Swarms with Applications in Medicine	94
8.4	A Statistical Physics Perspective on Human Dynamics Modeling	96
8.5	A Non-Equilibrium Approach to Biological Communication and Processing Theory .	97

List of Tables

4.1	NoC performance as a function of packet injection rate for a 4×4 mesh with input and output buffers of 20 and 10 slots, using XY wormhole routing and running an MPEG4 decoder (see Figure 4.1 for MPEG4 task graph details). Higher packet injection rates cause a decrease in the percentage of exponentially distributed inter-arrival times.	35
6.1	Commercial and scientific workloads used to test the constrained finite horizon fractal optimal control approach to power and peak temperature management.	57
7.1	Comparison between a non-fractal model (1-step ARMA) and a fractal model (see Eq. 7.10 in terms of the estimated parameters and the goodness-of-fit obtained for five time series of heart rate activity. The R-R interval time series are obtained for healthy individuals [126]. Except the time series with ID 3, all other hear rate activity series can be modeled through a fractional order differential equation of the type presented in Eq. 7.10. This is justified by the estimated p-values and test statistics.	87

List of Figures

1.1	Schematic plot of a 3×3 mesh Networks-on-chip (NoC) architecture consisting of 9 tiles. Any NoC tile consists of a processing element (PE), input and output buffers and an on-chip router through which the current PE communicates with the remaining PEs in the NoC.	3
1.2	a) A generic representation of the CPS operation from physical processes all the way up to application workloads. Such an itinerary may consist of volcanic activity monitoring, cloud movement, precipitation formation, air pollution, traffic conditions, and physiological processes, followed by compressing the data measurements and communicating them to data centers for further analysis. b) Feedback control diagram that ensures the QoS reference via statistical physics approaches. Distributed controllers can dynamically estimate the workload and decide, based on specific QoS metrics (<i>e.g.</i> , latency) on prioritizing data transmission or allocating more efficiently the communication bandwidth.	4
1.3	a) A segment as an example of an one-dimensional non-fractal object. b) The Cantor set as an example of a fractal object confined to one dimension. The Cantor set is obtained by dividing a segment in three equal parts and removing always the middle one. c) A square as an example of a non-fractal object characterized by a fractal dimension of 2. d) A branching tree object characteristic to natural systems and characterized by a fractal dimension of 1.8. e) Mass exponent $f(q)$ as a function of the moment index q . f) The multi-fractal spectrum $h(\alpha)$ as a function of fractal dimension α	9
1.4	Research objectives in this thesis: Building on statistical physics concepts, we propose an analytical framework for quantifying the NoC performance and fault-tolerance. To provide accurate optimization methodologies, we propose a master equation approach to traffic modeling. This serves as the basis for our future work on control and optimization for both NoC architectures and CPS systems.	11

2.1	a) CPU usage percentage as a function of time for a Genuine Intel (R) CPU running at 2.0 GHz with 2GB of RAM and a Microsoft Windows XP OS. During this period the CPU runs text editing, Matlab, web-browser and video applications. b) Variation in the CPU usage percentage between consecutive time stamps.	15
2.2	Probability of a CPU usage increment exceeding a given threshold (blue dots) is better fitted by a stable distribution (green line) rather than a Gaussian law (red line).	16
2.3	a) Virtual memory committed as a function of time for a Genuine Intel (R) CPU running at 2.0 GHz with 2GB of RAM and a Microsoft Windows XP OS. b) Available physical memory as a function of time for a Genuine Intel (R) CPU running at 2.0 GHz with 2GB of RAM and a Microsoft Windows XP OS.	17
2.4	Statistical physics description in both space and time of both application (<i>e.g.</i> , communication volume, execution time) and architecture (<i>e.g.</i> , frequency) domains.	18
3.1	a) The header inter-arrival times are measured at the West input buffer of node (1,2) on a 4×4 mesh NoC (see Figure 3.2) with input and output buffers of 3 and 1 slot, using XY wormhole routing with packets consisting of 15 flits and running a multi-threaded online transaction processing (OLTP TPC-C v3.0, IBM DB2 v8 ESE & Oracle 10g Enterprise Database Server) application. b) Probability of inter-arrival times to exceed a given threshold does not follow an exponential or gaussian distribution, but rather is better fitted by a stable distribution. In addition, one can observe that there exists a change in slope of the tail distribution which is a sign that the inter-arrival time stochastic process exhibits multiple fractal exponents.	24
3.2	Diagram of a 4×4 mesh NoC where the symbols L, N, E, W, and S stand for the local, north, east, west, and south router connections, and <i>in</i> and <i>out</i> suffixes represent the input and output directions.	26
3.3	Packet inter-arrival times at the North input buffer of node (1,2) on a 4×4 mesh NoC (see Figure 3.2) with input and output buffers of size 10 slots, under wormhole XY routing (packets consist of 5 flits) and running a multimedia application.	27
3.4	Snapshot of the NoC architecture at time t (a) and time $t + \delta t$ (b). Each NoC node consists of a processing element (PE), a router (R) and several input and output buffers corresponding to the number of its neighbors. The highlighted slots of the buffers represent the buffer occupancy at a particular time. For instance, between t and $t + \delta t$, one packet is injected from the PE at (2,2) to the local input buffer, one packet previously in the input buffer of the PE at (2,2) is routed towards node (1,2) and a new packet is received at West input buffer from node (1,2).	28

3.5 Snapshot of two NoC nodes as a random graph (RG) at time t (a) and $t + \delta t$ (b). Each node in the RG represents a buffer in the NoC (e.g., $B_{(2,2)}L_{in}$ denotes the local input buffer between the PE and the router at (2,2) mesh coordinates). The numbers on the arrows show the <i>IN</i> and <i>OUT</i> degree of RG nodes. As shown above, the <i>IN</i> and <i>OUT</i> degree of the $B_{(2,2)}W_{in}$ RG node is 1 and 0, respectively, at time t , and 2 and 1, respectively, at time $t + \delta t$. The unconnected RG nodes, having zero IN and OUT degree, correspond to buffers not included in the communication process yet. The difference between the IN and OUT degree at a particular time t shows the buffer occupancy.	29
3.6 Empirical cumulative distribution function of the fitness values obtained from two <i>IN</i> degree time series corresponding to the North input buffers of the nodes located at (1, 2) and (2, 2), respectively, for a 4×4 mesh NoC running an MPEG4 decoder application (see Figure 4.1) and an average packet injection rate per node of 0.029 packets per cycle.	31
4.1 MPEG4 decoder task graph [133] mapped onto a 4×4 mesh NoC architecture.	34
4.2 a) The buffer overflow probability as a function of packet injection rate for the input buffer to a memory block of an MPEG4 decoder mapped on a 4×4 mesh NoC with finite buffers (input and output channel buffer sizes of 20 and 10 slots, respectively) and XY routing scheme. With higher packet injection rates, the buffer overflow probability exhibits a nonlinear behavior which can be captured only by using a multi-fractal approach. b) The multi-fractal spectrum as a function of the fractal dimension of the header inter-arrival times at the North input buffer at nodes (1, 2) and (0, 1), respectively (see Figure 3.2 for the diagram of a 4×4 mesh NoC).	36
4.3 a) Time series of packet latencies between a source node at (3, 0) and a destination node at (2, 2) for four packet injection rates. b) Kurtosis of the packet latency as a function of the simulation time and packet injection rate. The exhibited variation in the kurtosis trend represents not only the presence of a non-Gaussian behavior, but also a non-stationary signature.	37
4.4 a) Probability of packet latency between nodes (3, 0) and (2, 2) to exceed a certain threshold for an MPEG4 decoder mapped on a 4×4 mesh NoC (see Figure 3.2) with finite buffers (input and output buffer sizes of 20 and 10 slots, respectively), wormhole switching (7 flits per packet), XY routing scheme and several packet injection rates. b) The multifractal spectrum as a function of fractal dimension of the packet latencies between source at (3, 0) and destination at (2, 2) and several packet injection rates.	38
4.5 Proposed traffic analysis methodology and its implications at application and architectural levels. Starting from the application and architecture characteristics, we build a dynamic probabilistic model of the network traffic which can be used for solving various problems such as topology synthesis, mapping, scheduling or routing.	40

5.1	Stochastic communication consists of a Gossip Round executed concurrently by all nodes. A packet received at the input ports is first CRC checked for information integrity. If the packet is successfully received, then its TTL parameter is decreased and the packet is copied to the Send_Buffer. According to the forwarding probability (p_f) the packet is duplicated, CRC checked (to prevent errors occurred during routing decision) and copied to the output buffers.	42
5.2	Mean hitting time (<i>i.e.</i> , time elapsed while propagating a packet from source $(1, 1)$ to destination (N, N)) comparison between SRW and BARW approaches for various mesh sizes $(2 \times 2, \dots, 20 \times 20)$. The results are conservative as we consider a best case scenario for SRW (<i>i.e.</i> , the packet is sent to a randomly chosen neighbor in each round) and an average case scenario for BARW (<i>i.e.</i> , packets are forwarded with 0.25 probability).	44
5.3	Possible events (<i>e.g.</i> , packet duplication, packet successful transmission, buffer overflow, and packet corruption) between two neighboring nodes under the stochastic communication protocol.	46
5.4	a) Time-dependent probabilities at destination $(1, 20)$ to receive 0, 1, and 3 packets from node $(1, 1)$ on a 30×30 mesh obtained via numerical analysis. The probability of receiving at least 1 copy is almost 1 after 23 communication rounds. b) Time-dependent probabilities for nodes $(1, 20)$ to receive 0, 1, and 3 packets from node $(1, 1)$ on a 30×30 mesh obtained via simulation.	49
6.1	Overview of our methodology for controlling fractal workloads in multi-core systems. We first input the information about the application (<i>e.g.</i> , task graphs, packet injection rates) and architecture (<i>e.g.</i> , topology, routing) and the set of costs and performance constraints that have to be satisfied by the NoC platform. Building on NoC workload measurements (<i>i.e.</i> , queue utilization, arrival and departure processes), we build a fractal model of the queue dynamics by estimating the fractional order of the time derivative, the utilization contribution parameter and the arrival and departure coefficients. Next, we use this identified model to design an optimal controller, which determines the optimal operating frequencies for the VFIs such that the performance constraints are satisfied.	52
6.2	Representation of j -th and l -th neighboring VFIs where each PE is set to run, if necessary, at its own frequency. The $x_j(t)$ variable represents the utilization of the interface queue between the j -th and l -th VFIs. The $y_i(t)$ represent the utilization values of the interface queue between the i -th PE and j -th VFI. Note that various colors of the tiles in the above NoC imply that each island can run at a certain frequency.	53

6.3	a) Utilization of the queues at tiles (0,0), (0,2), (1,1), (1,2), (2,2), (2,1), and (3,3) for a 4×4 mesh NoC running Apache HTTP webserver application. b) The variation of the operating frequencies for the routers at (0,0), (0,2), (1,1), (1,2), (2,2), (2,1), and (3,3) necessary to reach the reference values imposed on the utilization of all queues.	58
6.4	Comparison between the utilization of all queues in the uncontrolled case and the queue utilization for the case of the fractal optimal controller.	59
6.5	a) Utilization of the queues at tiles (1,1), (1,3), (2,0), (2,2), and (3,3), respectively, for a 4×4 mesh NoC running the Cholesky matrix factorization. b) The operating frequencies for tiles at (0,0), (1,1), (1,3), (2,2), (2,3) and (1,3) necessary for all queues in the network to meet the imposed constraints.	60
6.6	Comparison in terms of queue utilization between uncontrolled and the control case.	60
6.7	a) Representation of an $M \times N$ mesh NoC where the color of each tile shows that it belongs to a certain VFI. b) Representation of tile at (i, j) location on the mesh containing a PE which can run at a certain supply voltage and frequency, a router and five queues connecting it with the neighboring tiles.	62
6.8	Peak temperature displays a power law dependency (with exponent $\epsilon = 0.7997$) on the operating frequency for Intel E8500 dual core processor confirming the scaling law in Eq. 6.28.	65
6.9	Distribution of the identified parameters corresponding to the fractional order derivatives characterizing the queue utilization dynamics (see Eq. 6.14) for a 4×4 mesh NoC employing XY wormhole routing with input queue sizes of 15 flits and running a 16 node multithreaded web server (apache HTTP server v2.0) application. There is a slight variation in the distribution of fractional orders of the derivative as a function of the interval of the running application. This justifies the need for dynamically identifying the parameters of the model and then solving the same optimization problem with different coefficients as a function of the application phases for better power management.	72
6.10	a) Probability of the inter-events times between successive changes in queue utilization at north input queue of the node at (0, 1) to exceed a given threshold cannot be approximated by an exponential distribution. These inter-event times were obtained for a 4×4 mesh NoC employing XY wormhole routing with input queue sizes of 15 flits and running a 16 node multithreaded web server (apache HTTP server v2.0) application. b) Detrended fluctuation analysis of the inter-event times between successive changes in queue utilization at north input queue of the node at (0, 1) reveals a scaling exponent of 0.87 and confirms the existence of long-range dependence and fractal behavior in the dynamics of the queue utilization process.	73

6.11 a) Detrended fluctuation analysis of the inter-event times between successive changes in queue utilization at east input queue of the node at (1, 3) reveals a scaling exponent of 0.85 and confirms the existence of fractal behavior in the dynamics of the queue utilization process. b) Computed queue utilization for six queues at tiles (0, 0) and (2, 1) and a finite control horizon of 1000 cycles. The controllers determine the operating supply voltages and frequencies for each router in the network such that the queues of the nodes at (1, 1), (1, 2), (2, 1) and (2, 2) reach a queue utilization reference value of 0.1 and the queues of the remaining nodes reach a 0.07 reference value. The proposed optimal controller is able to stabilize the queues at (0, 0) and (2, 1) at a reference value of 0.07 and 0.1, respectively.	74
6.12 a) The variation of operating frequencies for the routers at (0, 0), (1, 1), (1, 2), (2, 1), (2, 2) and (3, 3) necessary to reach the reference values imposed on queues utilization b) The necessary supply voltage for the routers at (0, 0), (1, 1), (1, 2), (2, 1), (2, 2) and (3, 3) to meet the imposed queue utilization references.	75
7.1 CPS pyramid: The pyramid foundation summarizes the large set of physical processes of interest to the CPS community. Insights about physical processes gained through sensing, information fusion and aggregation lead to complex heterogeneous CPS workload. Accurate modeling of workloads enables the design of optimal CPS architectures that may improve our life. Similarly, to statistical physics which has been successful in explaining nature, the CPS workloads are modeled via master equations which later are used in defining various optimization problems of interest to the end users.	78
7.2 Graphical representation of transition probabilities assumed to govern the evolution of the stochastic process $x(t)$. In the absence of uncertainty, the transition probability from state $(x - y, t')$ can be neglected and the noise coefficient $\gamma(x - y) = 0$	81
7.3 Main challenges of CPS design: Resource allocation and network design to accurately sense a set of physical processes within a noisy environment, task mapping and computation scheduling to determine the optimal control strategies that allow satisfying the performance constraints.	83
7.4 A short electrocardiogram showing the heart activity of three beats and the P -wave (atrial depolarization), QRS complex (ventricles depolarization) and T -wave (ventricles repolarization). The $R - R$ interval is the time interval between two consecutive heart beats represented in the electrocardiogram by two consecutive R -waves. The variability observed in the $R - R$ intervals is essential for medical diagnosis.	84

7.5 Comparison between the measured $R - R$ intervals of an individual suffering from atrial fibrillation and the minimum and maximum bounds on the R-R intervals for a healthy individual at rest. One can clearly observe that the $R - R$ intervals have the tendency of exhibiting small magnitudes. For instance, the first 100 beats exhibit on average $R - R$ intervals of 0.55 seconds with a standard deviation of 0.14 seconds. This can lead to palpitations and so fainting and dizziness.	88
7.6 Applying the fractal optimal controller, the $R - R$ interval is increased from 0.19 to 0.80 (corresponding to a healthy heart rate of 75 beats per minute) in a finite control horizon of 0.1 seconds. b) Control signal - pacing frequency - necessary to be following by the optimal controller module of the pacemaker to increase the $R - R$ intervals and reduce the heart rate from approximately 100 to 75 beats per minute .	89
8.1 Dense network of micro-robotic swarms swim in hard to access regions of the body. The micro-robots dynamics is modeled as a collective behavior of multiple interacting random walks in a 3D graph tessellation of space. Each graph node has associated a binary random variable (σ) denoting whether or not it is occupied by one micro-robot. For detection and health monitoring purposes the goal is to find the time-dependent coverage of the swarm. The goal of the drug delivery problem is to find the probability for the nodes in the disease area (see Figure 8.1.b) to be occupied by micro-robots.	94
8.2 Hitting times for all micro-robots to reach their destinations shown for three interaction distances: $5 \mu m$ (5 hops) (a), $10 \mu m$ (10 hops) (b), and $15 \mu m$ (15 hops) (c). Distribution of hitting times is better approximated by means of the generalized extreme value, Fisk or Student's t-distribution rather than by means of Gaussian law when interactions are considered.	95

Chapter 1

Introduction

Due to tremendous advances in complementary metal-oxide-semiconductor (CMOS) technology, the computing paradigm evolved from its first incarnation (*i.e.*, ENIAC electronic computer) to complex Systems-on-Chip (SoCs) consisting of tens of heterogeneous cores seeking to address both application specific (*e.g.*, mobile phone communications, automotive, airplanes) and general purpose high-performance computing (*e.g.*, high energy physics, meteorology). The growing interest and demand for tera¹-scale (and soon peta²-scale computing) will soon require thousand-core SoC platforms. Nevertheless, moving deeper into the nanoscale domain under this multicore paradigm leads to a series of great challenges for SoC design [38]. Firstly, the increased wiring delays move the burden from computational resources to the communication side as driving long global interconnects increases the likelihood of synchronization errors due to unpredictable delays and high power consumption. Secondly, increasing the number of heterogeneous cores contributes to a higher design complexity. This has deep implications on both technological and time-to-market costs. Consequently, in order to mitigate such problems, the SoC design methodologies need to satisfy a few key attributes: scalability (*i.e.*, increased system size should lead to increased performance), re-usability (*i.e.*, increased re-utilization of components via decentralization and de-coupling of computation and communication), adaptability (*i.e.*, adaptation of computing platform to various workload and computational requirements), and reliability (*i.e.*, fault-tolerance to both transient and permanent hardware failures).

To tame the complexity of SoC design and provide enhanced scalability compared to traditional point-to-point or bus-based architectures (*e.g.*, long delays, area, power consumption), the design of electronic systems has to embrace the Network-on-Chip (NoC) approach [17][51][69]. Under the NoC paradigm, dedicated buses get replaced with packet based communication [50] allowing for a much higher communication bandwidth only with a moderate area overhead [36][77][92]. Nevertheless, the network concept, which stands at the heart of many natural and biological structures,

¹Tera-scale computing refers to platforms able to execute trillions (*i.e.*, 1 trillion = 1,000,000,000,000) of operations per second (teraflops) on trillions of bytes of data (terabytes) [39][68].

²Peta-scale computing refers to platforms able to execute quadrillion (*i.e.*, 1 quadrillion = 1,000,000,000,000,000) of operations per second (petaflops) on quadrillions of bytes of data (petabytes).

has become the *de facto* systems design paradigm. Consequently, it has become possible to build complex Cyber-Physical Systems (CPS) with the goal of reducing economical costs, minimizing energy cost and improving quality-of-life. Such CPS systems refer to both wired and wireless networks of embedded computational devices that can monitor and control various physical processes that occur in the environment (*e.g.*, a power grid, transportation and communication network, or network of medical devices).

To date, most NoC and CPS design and optimization methodologies assume either deterministic communication [21][75] or exponentially distributed events (*i.e.*, packet arrivals at buffers, cores, memory) with constant mean (or arrival) rates [46][61][117]. This implies that the probability of new event (*e.g.*, packet arrival, packet departure) decays exponentially fast to zero. Current models also rely on the memoryless property: the number of events appearing in disjoint time intervals are considered to be statistically independent [87]. These assumptions have important practical implications since Poisson traffic would require small buffers, predict small node-to-node latencies or other optimistic estimates for quality of service metrics. However, in reality, the network traffic is neither deterministic, nor stationary and/or memoryless.

Generally speaking, the network traffic is the result of a superposition of multiple correlated *dynamical processes* coming from the interaction between the user/application, on one hand, and the architecture/resources, on the other hand. Under such conditions, the highly nonlinear users behavior generates a complex workload dynamics that can exhibit a systematic relationship across multiple scales in space and time. At the same time, distributing computations over a large number of cores, makes it impossible to predict and inform all cores and resources about the global system state. This incomplete information about application dynamics as a function of user demands and the availability of computational resources calls for a new optimization methodology that considers both workload features (fractality, non-stationarity) and uncertainty in system behavior. This is the underlying theme of this PhD work.

1.1 Multicore Systems for Embedded and Cyber-Physical Applications

So far, we briefly learned about the increasing demand for more computing capabilities without justifying why this trend has been sustained and why it is believed to be sustained in the future. At last, applications are the main drivers of all information technology evolution and in turn they are subject to change, enhance in their increasing sophistication and complexity to address the challenges of our society at large. Computing platforms evolved from the classical Turing automata into complicated interactive Turing machines, and soon, to attain the societal needs, will go far beyond Turing paradigm. Consequently, in what follows, we review the main challenges for designing thousand core NoCs and then embark into the world of CPS aiming at integrating classical computing with surrounding physical processes for overcoming the challenges of the twenty first century.

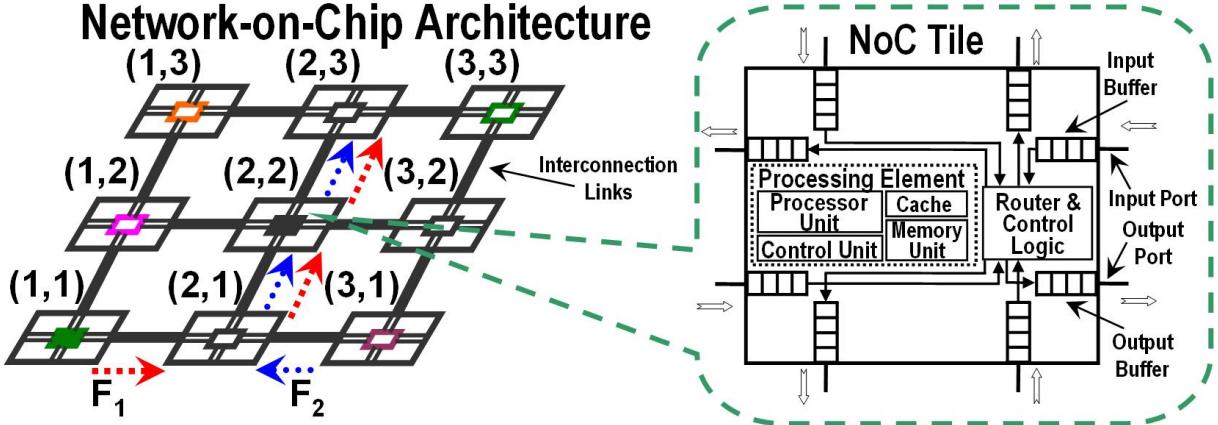


Figure 1.1: Schematic plot of a 3×3 mesh Networks-on-chip (NoC) architecture consisting of 9 tiles. Any NoC tile consists of a processing element (PE), input and output buffers and an on-chip router through which the current PE communicates with the remaining PEs in the NoC.

1.1.1 Network-on-Chip Architectures

The basic NoC architecture for multi-core platforms consists of a set of interconnected tiles that exchange packets among them. As shown in Figure 1.1, an NoC tile consists of a processing element (PE) (*e.g.*, general purpose or digital signal processor, graphics accelerator, embedded memory block), input and output buffers and an on-chip router through which the inter-tile communication is achieved. To support the inter-tile communication, each core has embedded input and output buffers to temporarily store the incoming packets from the neighboring nodes in the network. As shown in Figure 1.1, a packet generated at source (1,1) that needs to be delivered to destination (2,3) via a static XY routing, is first sent from local PE to its associated router at tile (1,1); then, at each intermediate node, a routing decision is made based on the header information (as shown with the red dotted arrows in Figure 1.1 for the shortest source-destination path).

Using the concept of packet based communication that avoids driving long wires, NoCs allow for a seamless integration of a large number of communicating cores; by proper design, this leads to a high system throughput/bandwidth. Besides providing scalability, the NoCs allow to design reusable and reconfigurable modular structures. In other words, while the PEs, routers, interconnects and network interfaces can all be designed and optimized in isolation, in the end they should be interconnected to build a complex computing platform that can be further reused to run a large and diverse set of target applications.

Despite these advantages, the NoC optimization is strongly dependent on the accuracy of traffic models [27][28][29]. Indeed, relying solely on stationary Markovian traffic models can lead to very optimistic estimates in terms of both buffer lengths or node-to-node latencies. As we show later in this thesis, due to the fractal and non-stationary nature of NoC traffic that results from the interaction between the user and application, on one hand, and architecture and the resources of the platform, on the other hand, the NoC optimization needs to rely on very different models compared to the queueing and Markovian models proposed so far [117]. This is particularly true

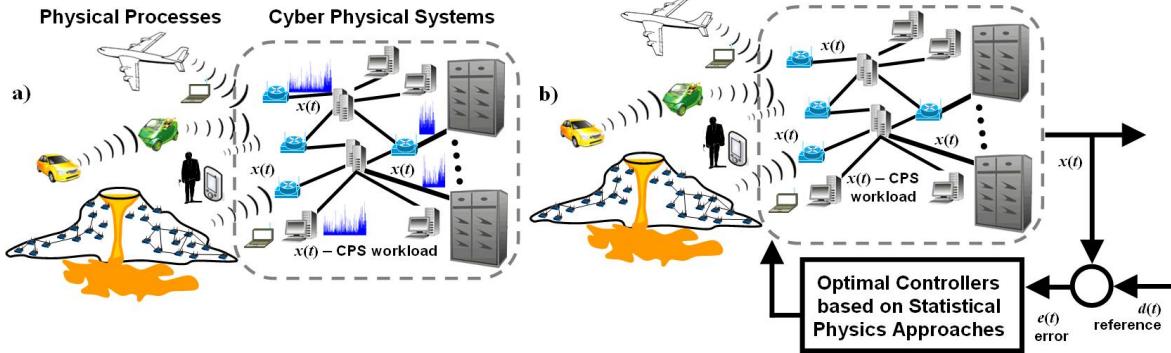


Figure 1.2: a) A generic representation of the CPS operation from physical processes all the way up to application workloads. Such an itinerary may consist of volcanic activity monitoring, cloud movement, precipitation formation, air pollution, traffic conditions, and physiological processes, followed by compressing the data measurements and communicating them to data centers for further analysis. b) Feedback control diagram that ensures the QoS reference via statistical physics approaches. Distributed controllers can dynamically estimate the workload and decide, based on specific QoS metrics (e.g., latency) on prioritizing data transmission or allocating more efficiently the communication bandwidth.

at high packet injection rates that push the network towards criticality (traffic congestion) [23][26].

1.1.2 Cyber-Physical Systems

Embedded systems research evolved to the point at which it enables tight integration of sophisticated sensors monitoring real world processes (e.g., heart rate, cloud movement, temperature fluctuations, wind speed, volcanic activity, earth magnetism fluctuations) and actuators able to control the environment. These newly envisioned systems, expected to successfully integrate computation, communication and control with physical processes, are commonly referred to as cyber physical systems (CPS) [93][94][147]. As shown in Figure 1.2.a, the CPS denotes a network of embedded computational devices and an associated set of wired or wireless networks that can monitor and control various physical processes that occur in the environment (e.g., a power grid, transportation and communication network, or network of medical devices) [31].

Unlike embedded systems research, where the focus is on building computational models for specific embedded applications, in the CPS area the goal is not only to establish a reliable communication infrastructure between such computational elements, but also to include *time- and feedback-based control* as intrinsic components of the programming model (see Figure 1.2.b) [30]. This goal lets us generalize the embedded-systems computational paradigm so that more-direct interaction between the system and physical world becomes possible. For instance, vehicular networks describing the cars movement in a city or the swarms of bacteria used for diagnostic or drug delivery purposes [32] are CPS examples that are distinct from classical networked embedded systems.

Building high performance, low power, as well as safe reliable and secure CPS requires a mul-

tidisciplinary design and optimization approach that brings together concepts, algorithms and techniques from real-time computing, signal processing, distributed and/or self-organizing control. The crux of these challenging problems is to rely on accurate workload models (see the $x(t)$ variable in Figure 1.2.b) since workload affects not only local parameters (*e.g.*, buffer utilization, core stall times), but also macroscopic metrics (*e.g.*, CPS throughput, CPS response time). For instance, we cannot decide the size and topology of a particular wireless sensor network without considering the spatio-temporal characteristics of the communication workload that must be communicated reliably to data centers for further analysis and decision purposes. In addition, we cannot arbitrarily decide the size of the communication buffers between the sensors in a network or data center because the loss or delay of critical information can have catastrophic effects on air, road, or railroad traffic. Similarly, we cannot ignore the characteristics of the workload generated by a series of bio-implantable devices, because this can have a crucial impact on a patient's life.

To overcome these challenges and in particular the problem of large scale, we argue for capturing the space-time characteristics of CPS workloads via fractal-type master equations and incorporate them into statistical physics (*e.g.*, mean field) inspired optimization framework (see Figure 1.2.b). This will be discussed in more details in Chapter 7.

1.2 Theoretical Foundations

Scientists are intensely preoccupied by the study of *dynamic processes taking place on networked architectures*, such as chemical pathways and gene regulation in biological systems, traffic across transportation networks like highways or flight trajectories, social networks, economic networks of finance and stock market or off- and on-chip technological networks. Although these dynamical processes have been considered random due to their complex and seemingly irregular behavior, they all share common universal characteristics such as *non-stationarity*, *non-ergodicity* and *fractality*. We should not simply be content with having gained an understanding of the characteristics of dynamical processes, but envision improved designs and optimization methodologies. This thesis relies on the fields of *statistical physics* and *fractional calculus*. It builds upon the theoretical foundations of science (*i.e.*, the interplay between hypothesis definition, experimentation and validity demonstration), in order to provide a comprehensive analytical framework that takes into account the optimization of various dynamical processes on networked architecture in the presence of fractality, non-stationarity and uncertainty.

1.2.1 Statistical Physics

A wide variety of systems within and surrounding us prove to be *out-of-equilibrium*: gases, fluids, superfluids, microtubules inside the cell, electrons in semiconductors, nuclear matter in neutron stars. Concerned with these out-of-equilibrium systems consisting of a large number of components (*e.g.*, particles, spins, molecules, agents), *statistical physics* describes the emergent macroscopic behavior resulting from both short- and long-range microscopic interactions (of many

degrees of freedom) [16][89][103][144].

Simply speaking, statistical physics overcomes the difficulties of classical mechanics (*i.e.*, writing dynamical equations of motion for individual erratically moving particle) by relying on probabilistic tools and metrics (*e.g.*, probability distribution function) to characterize population of agents and how they are affected by microscopic interactions in both space and time.

Since its birth more than a century ago, when scientists were studying the erratic movement of Brownian particles, statistical physics has found applications in a wide range of fields ranging from engineering [13][54], to biology [113], to economics [76][139] and social sciences [113]. In spite of many theoretical developments such as pattern formation, critical phase transitions, many particles systems or self-organization, relevant for the current discussion will be the master equation concept used to study the dynamics of buffer occupancy across a communication network, the spreading of a piece of information across a certain network and the transport processes (*e.g.*, pedestrian movement, car traffic) evolving over dynamic networked structures. All these subjects are discussed in more details in the next chapters.

The master equation refers to a differential equation which models, either in continuous or discrete time domain, the evolution of the probability of a given system to be in one state out of a set of many possible states. The derivation of master equation can be done using the very basic principles of probability theory, such as the total probability theorem. Building on such mathematical concepts, a generalized master equation meant to account for Non-Markovian dynamics can be written in the following form:

$$\frac{\partial P(x, t)}{\partial t} = \int_{t_0}^t \int_{-\infty}^{\infty} K(x - y, t - \tau) P(y, \tau) dy d\tau + P(x_0, t_0) \delta(x - x_0) \delta(t - t_0) \quad (1.1)$$

where $P(x, t)$ is the probability of finding the system of interest in state x at time t , $P(x_0, t_0)$ is the probability that the system started its evolution at time t_0 in state x_0 , and $K(x - y, t - \tau)$ is a kernel function of transition probabilities that weighs in the state space and in time the transitions from state y at an earlier time τ to the current state x at time t [110]. The generalized master equation above has been used to model many physical phenomena such as anomalous diffusion [83], continuous time random walks [65][107]. Assuming that our stochastic process $x(t)$ is Markovian, the generalized master equation degenerates into a classical Markovian equation of the form [98]:

$$\frac{\partial P(x, t)}{\partial t} = \int_{-\infty}^{\infty} K(x - y) P(y, t) dy + P(x_0, t_0) \delta(x - x_0) \delta(t - t_0) \quad (1.2)$$

where $K(x - y)$ depends only on the shifts in state space $x - y$ and not on time shifts. Such a Markovian equation has been employed by Bachelier (1900) to model price fluctuations, Smoluchowski (1906) for modeling density fluctuations in gas kinetic theory, Chapman (1928) for modeling the Brownian motion of grains in liquids, and Kolmogorov (1931) for his unified theory of continuous time Markov processes.

Throughout this thesis, we will make use of this master equations to study the network traffic processes and extend them to multiple coupled stochastic processes to model the spreading of branching random walks across a graph.

1.2.2 Fractional Calculus

There is good evidence that careful analysis of dynamical processes taking place on networked architectures, unveils that they display a rich spectrum of properties such as non-differentiability, non-Gaussianity, power law relaxation times or time fractality, which cannot be properly described via classical (integer order) calculus. The first “earth-shaking” moment, when dynamics ceased to be seen through the view of analytic functions, dates back to 1872, when Weierstrass constructed a function that was continuous, but nowhere differentiable. This mathematical artifact remained somehow a curiosity until 1960s when Mandelbrot [102] defined the fractals as the building block of explaining the natural geometries (*e.g.*, coastlines, lightnings, molecular trajectories, mountain and cloud shapes). Not long after that, Mandelbrot defined the notion of fractal time in the context of transmission in telephone networks [20] and model the fractal behavior via fractional calculus concepts [101].

Stemming from a scientific conversation between Marquis de L'Hôpital and Gotfried Wilhelm Leibniz in 1695 which scrutinized the existence and meaning of a half order derivative, *fractional calculus* refers to integral and derivative operators of fractional order. From a mathematical perspective, a fractional order derivative refers to a convolution of the considered function with a power law memory kernel. From this perspective, the stochastic dynamics exhibiting power law (rather than exponential) time correlations is more accurately modeled via a time fractional derivative of the probability distribution function [161]. The mathematical definition of the fractional order derivative is as follows:

$$\frac{d^\alpha x(t)}{dt^\alpha} = \frac{1}{\Gamma(n-\alpha)} \frac{d^n}{dt^n} \int_0^t \frac{x(\tau)}{(t-\tau)^{\alpha-n+1}} d\tau, \quad n-1 < \alpha < n, \quad n \in \mathbb{Z} \quad (n \text{ is integer}) \quad (1.3)$$

where α is the fractional order of the fractional derivative and $\Gamma(n-\alpha) = (n-\alpha-1)!$ is the Gamma function [128].

This continuous time definition of fractional derivative can also be written in discretized form via the Grunwald-Letnikov formula:

$$\frac{d^\alpha x(t)}{dt^\alpha} = \lim_{\delta t \rightarrow 0} \frac{1}{\delta t^\alpha} \sum_{j=0}^{[t-a]/\delta t} (-1)^j \binom{\alpha}{j} x(t-j\delta t) \quad (1.4)$$

where δt is the time increment, $[t-a]/\delta t$ represents the integer part of the ratio between the $(t-a)$ and δt . Eq. 1.3 (continuous) and Eq. 1.4 (discrete) capture directly the role of the power law observed in the intervals of times between which the stochastic process changes its magnitude (*i.e.*, $(t-\tau)^{\alpha-n+1}$) and allows not only for a more accurate description of the time dynamics of stochastic

process $x(t)$, but also better optimization as will be discussed in this thesis.

Besides Mandelbrot's fractional Brownian motion model, fractional calculus has found applications in a wide range of fields such as strange kinetics [165], viscoelasticity [99], stochastic dynamics with long-time memory [161], bio-physiological processes [64][162], anomalous diffusion [44][45][84][85][123], field dynamics [151], reaction-transport systems [106], control engineering [8][119][127][2], non-extensive statistical mechanics [1] and economics [72]. In this thesis, fractional calculus concepts are used to model fractal processes and define new optimization methodologies for both NoC architectures and CPS systems.

1.2.3 A Glimpse into Fractals

Although self-similarity was debated and discussed by many mathematicians and physicists between 1600s and 1960s, the concept of fractal was introduced in 1975 by Benoit Mandelbrot to denote an object that exhibits a jagged shape or spiky behavior and some degree of repetition or self-similarity over a wide range of scales.

Mathematically speaking, a geometric object is called *fractal* if it displays self-similar characteristics over all scales and it is characterized by a fractional dimension:

$$\text{fractal dimension} = \frac{\text{number of self-similar pieces}}{\text{magnification factor}} \quad (1.5)$$

To better understand what fractals (or non-Euclidean geometry) mean, we consider the side by side comparison of four geometric objects: a straight line (see Figure 1.3.a), the Cantor set (see Figure 1.3.b), a square (see Figure 1.3.c) and a collection of tree branches (see Figure 1.3.d). Figure 1.3.a shows a segment of arbitrary length which can be divided into an arbitrary series of smaller segments of equal length. This shows that this one dimensional Euclidean object displays self-similar properties. By computing the fractal dimension via Eq. 1.5 we obtain an integer value of 1 for any magnification factor. By following the same reasoning, we can deduce that the fractal dimension for a square is 2 irrespective of the magnification factor. This implies that some Euclidean objects may display self-similar structure, but are not fractal since they are characterized by integer dimensions.

In contrast, when computing the fractal dimension of a Cantor set we obtain a fractional value of approximately 0.6. For completeness, we consider a collection of tree branches and use the box covering method to determine a fractional dimension of 1.8. Both these objects, not only display self-similar features, but are also characterized by fractional dimension coefficients. In conclusion, these geometric objects can be regarded as fractals.

In addition to these examples, there are many natural objects that display self-similar features but characterized by different fractal dimensions. These objects represent a generalization of the fractal concept and are called *multi-fractals*. Consequently, unlike fractal objects which are characterized by a single fractal dimension, the multi-fractal objects are characterized by a distribution of fractal dimensions.

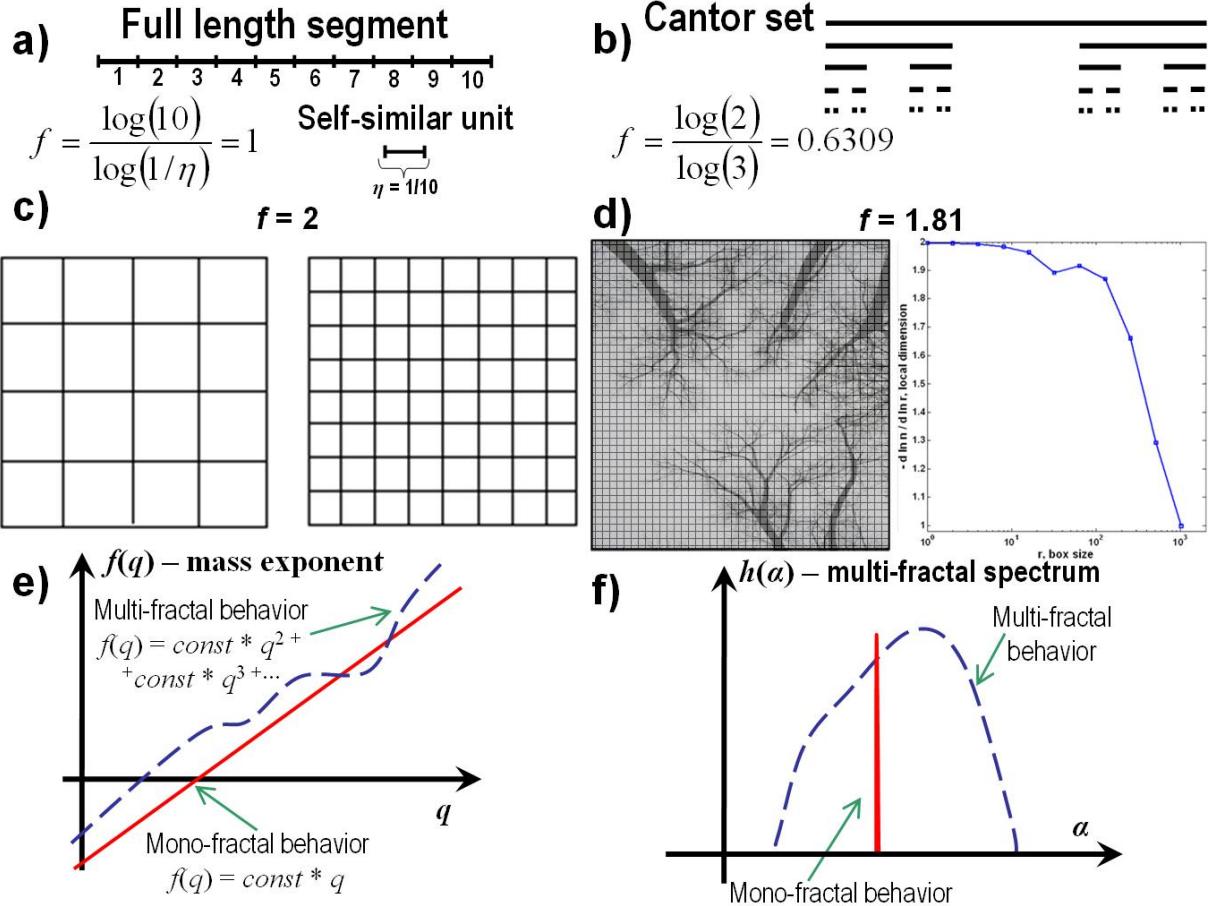


Figure 1.3: a) A segment as an example of an one-dimensional non-fractal object. b) The Cantor set as an example of a fractal object confined to one dimension. The Cantor set is obtained by dividing a segment in three equal parts and removing always the middle one. c) A square as an example of a non-fractal object characterized by a fractal dimension of 2. d) A branching tree object characteristic to natural systems and characterized by a fractal dimension of 1.8. e) Mass exponent $f(q)$ as a function of the moment index q . f) The multi-fractal spectrum $h(\alpha)$ as a function of fractal dimension α .

As a result of availability of many rich data sets about dynamical processes, the geometric interpretation of fractals has been extended to probability theory and stochastic processes [18]. Stochastic fractals gained a lot of attention in statistical physics for modeling anomalous diffusion or Levy flights [84][80], transport processes [85], growth models [15], disordered systems [45], molecular dynamics [64], reaction limited aggregation [16], physiology [124]. The discovery that price fluctuations are not completely random lead to a very intense investigation of the existence of fractal behavior in econophysics [102][100]. More recently, fractals have been discovered in complex networks [54][113] and complex systems [125][144].

From a mathematical perspective, a stochastic process is denoted as being mono-fractal if all its moments of order q behave as $K_q(t) \approx t^{\text{const} \times q}$. As shown in Figure 1.3.e by red straight line, the exponent of the higher order moments $K_q(t)$ is called mass exponent $f(q) = \text{const} \times q$ and

is linear in the order q . However, in many situations, due to the heterogeneity in the system or the interactions between the system components and environment, this mass exponent function $f(q)$ turns out to behave as a nonlinear function of the moment order q (*i.e.*, $K_q(t) \approx t^{f(q)}$, where for instance $f(q)$ can be given by $\sqrt{q} + q^2$). In this case, the stochastic process is called multi-fractal (see the blue dotted line in Figure 1.3.e). An alternative description of fractal processes can be obtained by computing the distribution of fractal dimensions. As shown in Figure 1.3.f, a fractal stochastic process is characterized by a peaked (delta-function type of law) distribution. In contrast, the multi-fractal stochastic process is characterized by a wide distribution of fractal dimensions (see the blue dotted line in Figure 1.3.f). All these concepts are used in this thesis to model dynamical processes taking place on a networked infrastructure and then based on these models we define various optimization methodologies.

1.3 Thesis Overview and Research Objectives

Although promising from several points of view (*e.g.*, design modularity, higher scalability, increased energy efficiency per bit operation), the NoC architectures need to be able to communicate reliably large amounts of data resulting from various applications (*e.g.*, biological and chemical-physics simulations, real-time 3D Internet, media mining, health monitoring and interactive medical diagnosis, financial analysis, *etc.*) in order to deliver the promise of tera-scale computing. For achieving maximum performance, a new science of network design is needed to account for the dynamic nature of incoming applications and efficiently map them onto the available computational resources. Towards this end, the proposed research addresses the following major objectives:

- **Modeling and analysis of dynamic processes taking place on networked architectures:** Because the performance of NoC architecture is highly dependent on the packet injection rates, routing protocols and the amount of buffering resources available on-chip, we develop a *statistical physics inspired NoC traffic* model able to capture the dynamic characteristics (*e.g.*, fractality, non-stationarity) of traffic patterns [23][28]. As we show in Chapter 3 and 4 (see also left hand side of Figure 1.4), this traffic model is crucial for performance analysis (*e.g.*, buffer occupancy, node-to-node latency) and platform optimization (*e.g.*, buffer sizing, packet scheduling).
- **A dynamic approach to NoC fault-tolerance:** Relying on the *master equation* concept, we model the *dynamic processes* (packet duplication, packet probabilistic dissemination, packet dropping) enabled by a stochastic communication protocol and estimate the hitting time between a pair of source and destination nodes [25]. As shown in Figure 1.4, this formalism can not only be used to estimate the performance of NoCs in the presence of hardware errors, but also to define several optimization problems (*e.g.*, find the forwarding probability such that the hitting times of source-destination pairs and the total number of disseminated packets are minimized).

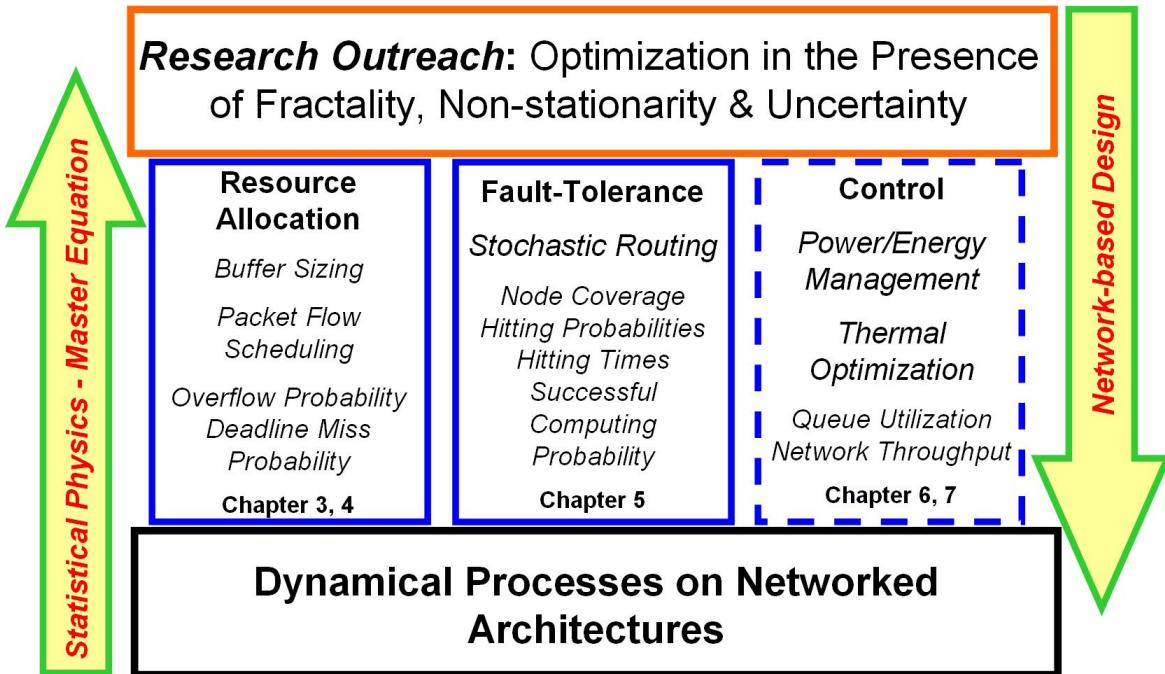


Figure 1.4: Research objectives in this thesis: Building on statistical physics concepts, we propose an analytical framework for quantifying the NoC performance and fault-tolerance. To provide accurate optimization methodologies, we propose a master equation approach to traffic modeling. This serves as the basis for our future work on control and optimization for both NoC architectures and CPS systems.

- **Optimization of networked architectures under fractality, non-stationarity and uncertainty:** Building on our traffic model, our next step is to propose a *new optimization methodology* for NoCs which takes into account the traffic characteristics and its intrinsic uncertainty (see the right hand side of Figure 1.4). More precisely, we aim at solving a *finite horizon optimal control problem* that seeks to keep the buffer utilization under a predefined region while satisfying some performance constraints (*e.g.*, latency) and minimizing power consumption and/or peak temperature profile.

All in all, we embrace a statistical physics approach to characterize the dynamic processes taking place on NoCs via a master equation. This equation characterizes the dynamic processes in both space and time and thus allows us to compute their higher order statistics. In addition, this methodology also offers an elegant way of formulating and solving various NoC optimization problems (*e.g.*, power management).

Building on our modeling framework of dynamical processes taking place on networks, we propose a general optimization approach for cyber-physical systems. Although this formalism can be easily particularized to classical problems such as the resource allocation under fractal workloads, we focus our attention on pacemaker algorithms for controlling fractal heart rate processes. We also generalize our analytical framework to capture the human dynamics through fractal structures

(*e.g.*, road traffic) or to model biological swarms in non-Newtonian³ liquids.

1.4 Thesis Organization

This dissertation focuses on modeling, analysis and optimization of dynamical processes taking place on networked structures. Unlike classical queueing and linear control theory, we account for the processes characteristics like fractal and non-stationary behavior and present novel design methodologies and design automation tools for power and thermal management of future NoC-based multi-cores. In the following, we provide a brief overview of our contributions.

Chapter 2 sets forth a (non-equilibrium) statistical physics description of both application and architecture domains. More precisely, it models the set of applications running on a computational architecture as a dynamic graph of interacting computational tasks that grows and shrinks as a function of user preferences, operating system requirements and architectural constraints. Note that we do not only consider the time as an intrinsic component of current and future interactive computation, but also account for fractal behavior and uncertainty by characterizing graph nodes, interaction edges and their computational properties via time dependent probability distribution functions. Along the same lines, we introduce a novel time-dependent probabilistic description of architectural components. This is motivated in part by the intrinsic process variability, manufacturing non-idealities and susceptibility of hardware to particle strikes which can affect the operation either transiently or permanently.

Building on this non-equilibrium description of both application and architecture, Chapter 3 introduces a statistical physics inspired approach to network traffic modeling. More precisely, the non-stationary and fractal behavior observed in network traffic are modelled through a generalized master equation. In addition, we introduce the concept of fitness distribution in order to account for the heterogeneity that can exists in network traffic patterns. We also discuss the main differences between our approach and the existing traffic models for multi-core platform design.

Starting from this statistical physics characterization of network traffic, Chapter 4 summarizes the derivation of the differential equation governing the evolution of higher order moments associated with network traffic processes and investigates the implications of fractal behavior on performance (*e.g.*, buffer overflow probabilities, source-to-destination exceedance probabilities). In addition, we investigated the critical phenomena in NoC traffic by quantifying the departure from Markovian assumptions as a function of increased packet injection rate for synthetic traffic and MPEG4 decoder application. For completeness, we validated our proposed NoC traffic model for both synthetic traffic traces and benchmark applications from SPEC 2000 and SPEC 2006.

Chapter 5 summarizes our work on modeling the stochastic communication protocol as a collection of branching and annihilating random walks constrained to move along the edges of the network topology. Besides this theoretical challenge, the emphasis is on capturing all network communication transactions and estimating the hitting time between a source and a destination node.

³Non-Newtonian fluids exhibit a nonlinear relationship between the shear stress and shear rate (unlike Newtonian fluids which exhibit a linear relation).

Later on, in Section 8.3, we discuss how this formalism can be used to model and quantify the performance of bacteria propelled micro-robotic swarms swimming in viscous environments.

Building on fractional calculus concepts to model the network fractal behavior, Chapter 6 presents a constrained finite horizon fractal optimal control approach for both power management and peak temperature minimization. To test the efficiency of our dynamic optimization methodology, we used benchmark applications from SPECweb99 and SPEC 2000. In addition, we also discuss how the parameters of the model can be efficiently estimated and how the controller can be synthesized at run-time from solving a linear system in a parallel fashion.

The aim of the proposed statistical physics inspired network traffic model is not only to model and reproduce the NoC traffic traces, but also to provide intuitive insights on how to perform better design and optimization of networked architectures. Along these lines, Chapter 7 builds on a fractional calculus based master equation and sets forth a general dynamic optimization approach for cyber-physical systems. As a concrete example of this formalism, we summarize our results on designing control algorithms for regulating fractal processes with predefined deadlines.

Finally, in Chapter 8 we outline our future work on three directions: Firstly, we seek to propose a coherent framework for modeling, analysis and optimization of biological propelled micro-robotic swarms with medical applications. Secondly, by coupling the description of agent dynamics through a fractal structure with a generic stochastic utility function, we formulate a new fractal dynamic game aiming to mitigate the modeling and analysis of human dynamics (*e.g.*, human crowds, road traffic, financial markets). Thirdly, building on the statistical physics formalism presented in this thesis and taking into account on one hand the heterogeneity that exists in biological populations and the biological characteristics such as aging, we suggest how growth models for stem cells can be developed.

Chapter 2

A Dynamic Perspective on Networked-based Architectures Design

2.1 The Big and the Small: Technology Implications on NoC Design Flow

One can easily and optimistically assume that we could formulate physical laws that describe the dynamics of an entity (*e.g.*, particle, spin, phase) in isolation with high degree of precision. In addition, we can also picture the successful description of two such entities interacting at various scales in space and time through the language of quantum mechanics. Nevertheless, as we go deeper and consider more and more entities and more and more types of long-range interactions rather than short-range interactions (which are the norm of Boltzmann-Gibbs statistical mechanics at thermal equilibrium) we discover that our physical laws and abstraction systems become insufficient to offer complete understanding and lack prediction capabilities. Obviously, keeping the discussion in abstract terms, we can reduce the entire phenomena to a collection of dynamical processes taking place on networked structures. In reality, for such phenomena there is the concept of condensed matter physics that has implications at both nano- (we refer to information switching devices build at nanoscale or consisting of a relatively small number of atoms) and tera- (we refer to low energy and fast information processing architectures consisting of thousand and more computing modules) worlds. In what follows, we review how information technology evolved and how the current work copes with the challenges we face in designing and optimizing future information processing systems.

The advances we witnessed over the last 60 years in electron charge based information processing systems may soon come to an end. Not only that it is difficult to draw and follow the device geometry at $1.4 - 5 \text{ nm}$ that can run at 40 fs , but it becomes highly expensive in terms of power density (*e.g.*, at least 100 W/cm^2) [168]. To overcome these technological challenges,

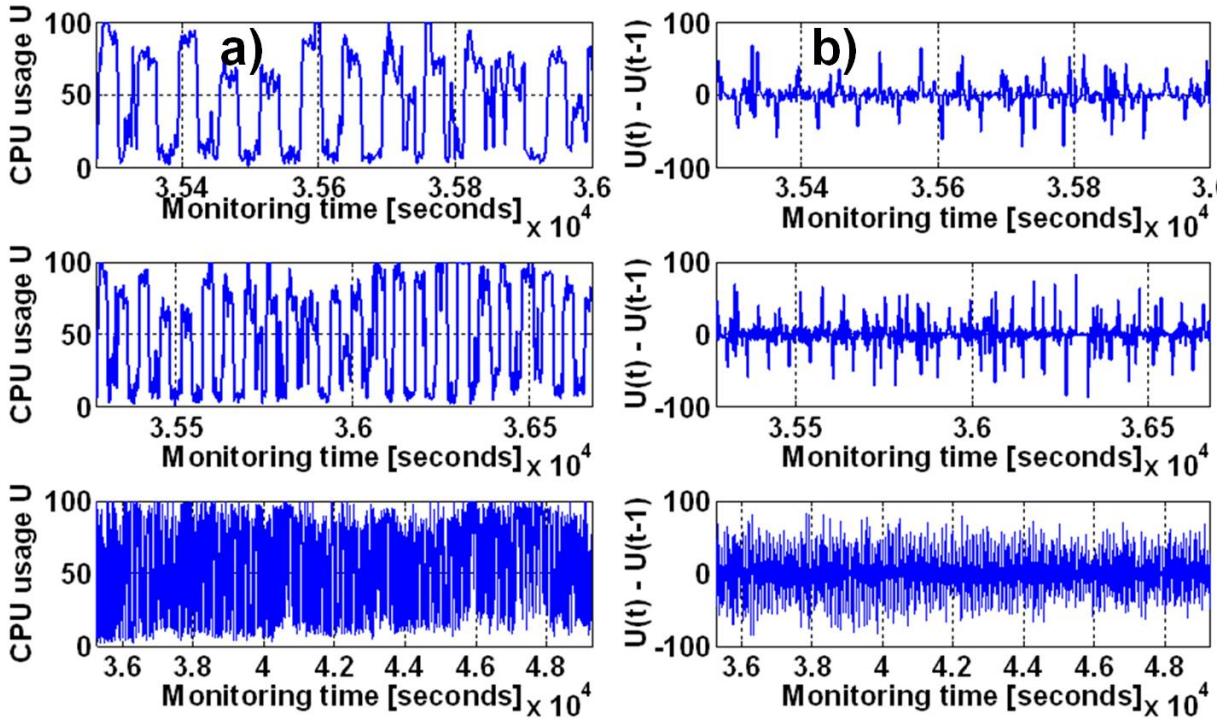


Figure 2.1: a) CPU usage percentage as a function of time for a Genuine Intel (R) CPU running at 2.0 GHz with 2GB of RAM and a Microsoft Windows XP OS. During this period the CPU runs text editing, Matlab, web-browser and video applications. b) Variation in the CPU usage percentage between consecutive time stamps.

the information processing switches need to cease relying on electron charge states and embrace alternative states of representing information. Condensed matter approaches not only lead to the discovery of such new information storage devices (*e.g.*, ferroelectric, ferromagnetic) but also offers hints about how a new theory of dynamical processes evolving onto a networked structure should be constructed when the equilibrium thermodynamic assumption needs to be forgotten. In addition, the need for a solid theory of dynamical processes on networked structures is even more needed when considering that many such nano-scale devices display a complex fractal structure in both space (chemical bond geometry) and time (microscopic memory resulting from fluctuations in energy barriers) [134], long-range interactions, non-homogeneous mixing to name just a few. Capturing the time-dependence and fractal behavior as two main characteristics of the operation of future information processing devices should not only allow accurate estimates about the device switching time or energy consumption, but also offer reliable predictions about the device error probability and even more so enable macroscopic architectural predictions that can prove useful at system level design and optimization.

Besides the out-of-equilibrium signatures exhibited at nanoscale, the current and future applications and architectures will continue to exhibit almost the same type of characteristics: computational and memory requests will likely be time dependent, availability of physical resources such as computational power or physical memory will be time dependent too, and above all their

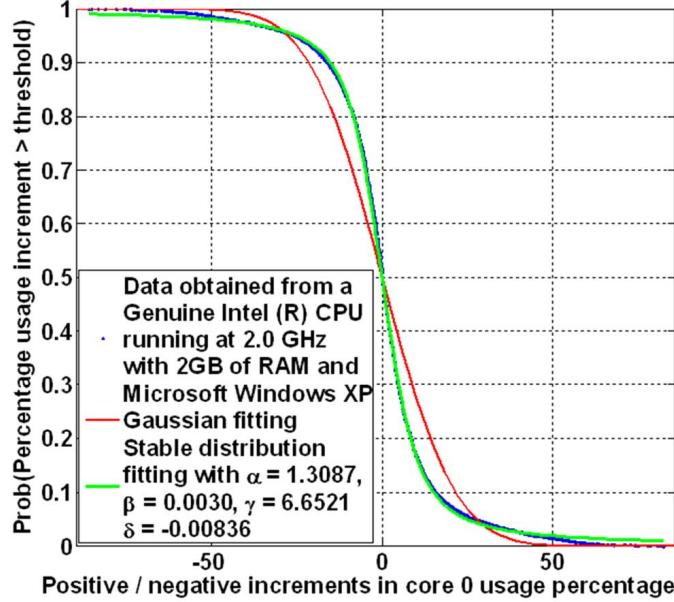


Figure 2.2: Probability of a CPU usage increment exceeding a given threshold (blue dots) is better fitted by a stable distribution (green line) rather than a Gaussian law (red line).

stochastic dynamics will not be completely random at all. To give a sense about the existence of time-dependence, self-similarity and/or non-Gaussian behavior in current computing platforms, we report in Figure 2.1 and Figure 2.2 the percentage of CPU usage as a function of time and in Figure 2.3 the physical memory availability as a function of time and more importantly user preferences regarding the pool of running application.

The first three vertical plots in Figure 2.1.a show the percentage of CPU usage as a function of time at three scales: hundred, thousand and tens of thousand of seconds. Overall, besides the persistent time-dependent behavior, the precentage of CPU usage does not look neither completely periodic nor completely random. When looking in Figure 2.1.b at the differences between percentage of CPU usage at two consecutive time stamps, we notice that the entire dynamics displays some kind of fractality which can come from correlations that can exists in both user behavior activities and different application phases.

Even more so, when analyzing the empirical probability of observing a difference in percentage of CPU usage between two consecutive time stamps to exceed a given threshold, we observe that it is better fitted by a stable distribution (closer to Holtsmark distribution¹) rather than a Gaussian law (see Figure 2.2).

Besides these patterns observed for computational dynamics, the communication processes also exhibit a time-dependent behavior as can be seen from Figure 2.3. More precisely, Figure 2.3.a shows the virtual memory committed as a function of time showing that memory operations coming from both application and operating system exhibit a pronounced time-dependent behavior. The

¹Holtsmark law is a particular case of a stable distribution with shape parameter $\alpha = 1.5$ and skewness parameter $\beta = 0$. Besides the fact that it is an example of a power law, Holtsmark distribution characterizes the fluctuations in plasma created by charged particles [84]

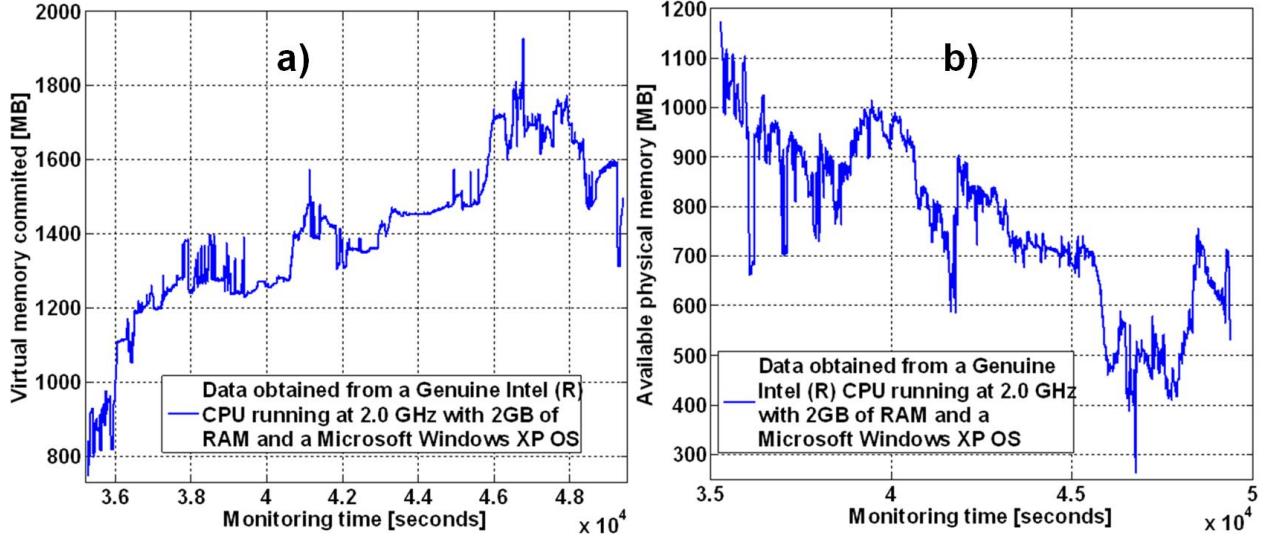


Figure 2.3: a) Virtual memory committed as a function of time for a Genuine Intel (R) CPU running at 2.0 GHz with 2GB of RAM and a Microsoft Windows XP OS. b) Available physical memory as a function of time for a Genuine Intel (R) CPU running at 2.0 GHz with 2GB of RAM and a Microsoft Windows XP OS.

availability of physical memory is time-dependent also (see Figure 2.3.b). This calls for new abstraction models of both application and architecture which should incorporate time as an essential component (resulting from the interaction between computing platform with the outside world) and the stochastic patterns as a way to develop accurate models enabling dynamic (self-organized) optimization.

Consequently, in the remaining of this chapter, we summarize a few statistical physics inspired definitions of both application and architecture domains by taking into account the dynamic nature of various stochastic processes (associated to packet generation and link/router failures) via time-dependent probability density functions (PDFs) as shown in Figure 2.4.

2.2 Application Modeling

Definition A *statistical physics application characterization graph* (SPACG) described by

$$App \left\{ P(v_i, t), P(e_{ij}, t), P(\lambda_{v_i}, t), P(\mu_{v_i}, t), P(\text{comm_vol}_{e_{ij}}, t), P(\epsilon_{v_i}, t), P(d_{v_i}, t) \mid v_i \in V_{App}, e_{ij} \in E_{App} \right\} \quad (2.1)$$

is defined as a dynamic (evolving) graph, where:

- V_{App} is the set of vertices, *i.e.*, each vertex $v_i \in V_{App}$ denotes a computational module of the application referred to as a computational task;
- $E_{App} \subset V_{App} \times V_{App}$ is the set of edges, *i.e.*, each directed arc $e_{ij} \in E_{App}$ represents a possible

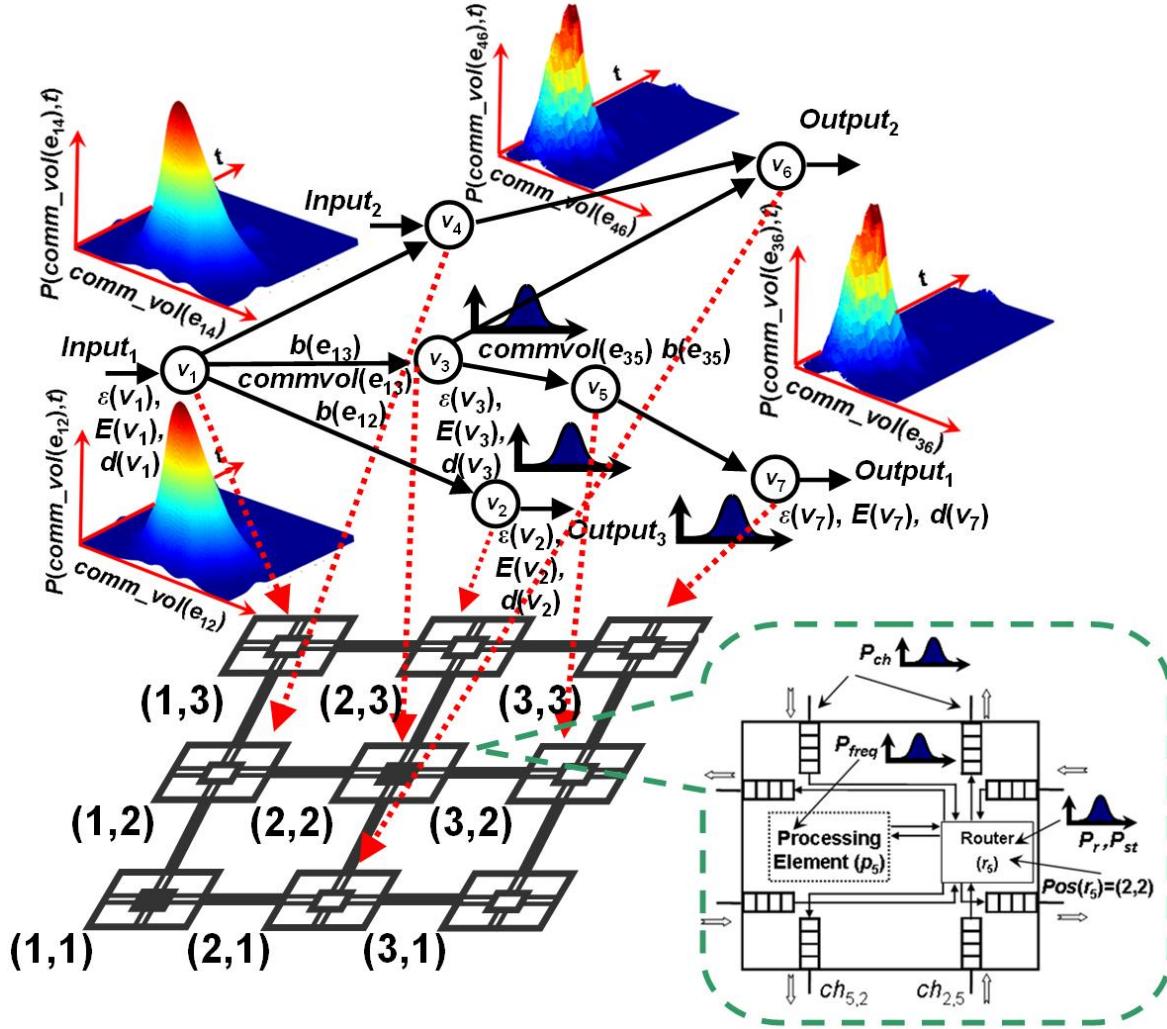


Figure 2.4: Statistical physics description in both space and time of both application (e.g., communication volume, execution time) and architecture (e.g., frequency) domains.

communication event (seen as a stochastic process);

- $P(\lambda_{v_i}, t)$ and $P(\mu_{v_i}, t)$ are the probability distributions associated with the stochastic process of packet generation and consumption at any vertex $v_i \in V_{App}$;
- $P(\text{comm_vol}_{e_{ij}}, t)$ is the communication volume (see Fig. 2.4) from vertex $v_i \in V_{App}$ to $v_j \in V_{App}$ (also regarded as a stochastic process);
- $P(\epsilon_{v_i}, t)$ is the probability distribution of the execution/computation time at vertex $v_i \in V_{App}$, and
- $P(d_{v_i}, t)$ represents the probability distribution function of the deadlines at each vertex $v_i \in V_{App}$ (as shown in Fig. 2.4).

Note that in relationship 2.1 the application is seen as a dynamic graph where nodes and edges

appear and disappear at execution. The appearance of edges may be triggered by that fact that a given running process on a multicore platform may instantiate a child process, access a shared library, read/write into a memory location or communicates data and control variables with another process. The disappearance of edges may signify that the set of running processes are terminated as a result of computation completion, kill process operation initiated by users or operating system. The entire dynamics of the graph structure is captured via the probabilities $P(v_i, t)$ (*i.e.*, the probability that node v_i is active at time t) and $P(e_{ij}, t)$ (*i.e.*, the probability that the edges e_{ij} exists between nodes v_i and v_j , respectively, at time t).

2.3 Networks-on-Chip Architecture Modeling

Future information processing devices not only at switch level, but also at core level will display a high degree of variability and will likely need to be seen and modeled as non-equilibrium systems (see the ITRS 2010 [142] and [159]). In future thousand core platforms, the inter-processor variability will play a crucial role not only because it affects the information processesing performance, but even more so it influences the overall energy consumption. Simply speaking, manufacturing and physics imperfection cause the less efficient processors to burn more power by requiring high supply voltages and lower threshold voltages in order to meet a certain performance level. Considering these physics based realities, we define next a statistical physics inspired (*i.e.*, time dependent probabiltic framework) description of the NoC architectures.

Definition The *Network-on-Chip Architecture* (NoC_Arch) is defined by the following tuple:

$$Arch \left\{ T(U, F), \Omega, S_P, P(r_i, t), P(ch_{ij}, t), P(freq_{p_k}, t) \mid r_i \in U, ch_{ij} \in F, p_k \in S_P \right\} \quad (2.2)$$

where the components have the following meanings:

- The *network topology* is defined as a labelled graph $T(U, F)$, where the routers ($r_i \in U$) and channels ($ch_{ij} \in F$) in the network are given by the sets U and F, respectively, as follows:
 - $\forall ch_{ij} \in F, w(ch_{ij})$ gives the channel bandwidth;
 - $\forall r_i \in U, l(r_i, ch_{ij})$ gives the buffer size (depth) of channel ch_{ij} , located at router r_i ;
 - $P(r_i, t)$ is the time dependent probability of failure of router r_i as shown in Fig. 2.4;
 - $P(ch_{ij}, t)$ is the time dependent probability of successful transmission of a packet routed at r_i via channel ch_{ij} ;
 - $P(st_{r_i}, t)$ gives the distribution of the packet service time (st) at router r_i at time t ;
 - S_P represents the set of processing elements (PEs);
 - $freq_{p_k}$ is the operating (clock) frequency of each PE $p_k \in S_P$;

-
- $P(freq_{p_k}, t)$ is the probability density function of operating frequencies as shown in Fig. 2.4;
 - $\Omega : S_P \times U \rightarrow \{0, 1\}$ is a function that maps a PE $p_k \in S_P$ to a router $r_i \in U$ (*i.e.*, $\Omega(p_k, r_i) = 1$).

Note that the network topology $T(U, F)$ does not need to be fixed; instead, it is likely that in the future links, buffers and routers can be reconfigured to meet the application and user requirements while minimizing energy consumption. In addition, note that the intrinsic characteristics of the stochastic dynamics of basic information processing devices (*e.g.*, the variability of the energy barrier in either charge/spin-based or molecular transistor) impacts the properties of the entire system; this is the motivation for introducing the time dependent probabilities for characterizing the router failure, router service time, link successful transmission, and/or the core operating frequency.

Given this time-dependent probabilistic description of the architectural components, the next step is to formalize the communication paradigm: the routing protocol through which the data and control packets are routed from sources to destinations.

Definition The communication paradigm

$$\Re \left\{ R(r_i, ch_{ij}, r_{src}, r_{dst}, \rho(t), t), Sw | r_i, r_{src}, r_{dst} \in U, ch_{ij} \in F \right\} \quad (2.3)$$

consists of the routing policy $R(r_i, ch_{ij}, r_{src}, r_{dst}, \rho(t), t)$ at router r_i , for a source router r_{src} and a destination router r_{dst} ; $\rho(t)$ denotes the utilization of channels connected to the neighboring routers at time t , and finally, Sw specifies the packet switching techniques (*i.e.*, a protocol to forward a flit through the channel $ch_{ij} \in F$ of router $r_i \in U$ towards the router $r_j \in U$). Note that static routing does not require the utilization information, while adaptive or stochastic algorithms may check the congestion of the immediate routers and pick the ones having the smalles utilization $\rho(t)$ at time t .

2.4 Challenges of Designing Thousand Core Network-based Architectures

Given that the availability of limited resources (such as those on supply and threshold voltages due to power and reliability concerns) has shifted the focus in computing from single high complexity processor designs to platforms consisting of thousand power efficient cores running high performance massively parallel programs and few complex cores dedicated to high performance sequential code parts. In addition, it is now accepted that all these cores will exchange data and instructions via a network environment rather than a P2P or BUS approach. Although the mirage of building such multicores starts to become a tangible reality and its applicability is fastly foreseeable in cloud computing or cyber physical systems, it also brings to the discussion many great challenges.

Within the framework of network based parallel architectures, it is not enough to size the network links and buffers, optimize routers and network interfaces, but also find out the right

memory hierarchy (*e.g.*, size of memory components), grant access policies and the strategies for maintaining memory coherency.

Related to the above mentioned issues, it is important to rely not only on accurate models of the applications, but also on the capabilities of various heterogeneous processors such that we can find the optimal² number of processors on which a set of applications is mapped and scheduled to run. In addition, it is important to be able to modify the routing protocols such that a minimum latency transfer is achieved between pairs of distant sources and destinations while considering the power budget and reliability issues.

Last but not least, designing and optimizing the operating system that connects the application and architecture domains needs accurate abstraction models to decide what the internal data structures should look like, how the partitioning, clustering, mapping, scheduling and migration of tasks and threads should be done, and when the managing strategy of resources to achieve high performance with minimum energy requirements should be changed.

In addition, the design and optimization of future thousand core Network-on-Chip architectures *cannot* be done by isolating individual components and tweaking each to achieve some performance level. In contrast, the future NoC design should be regarded as a global optimization approach which relies on accurate analytical models of both spatial and time components of the design space exploration. Towards this end, we formulate the the problem of stochastic network design as follows:

$$\begin{aligned}
& \min / \max \quad O(T(U, F), \Omega, S_P, P(r_i, t), P(ch_{ij}, t), P(st_{r_i}, t), P(freq_{p_k}, t), M, S, R) \\
& \quad \text{such that for a set of applications} \\
& App(P(v_i, t), P(e_{ij}, t), P(\lambda_{v_i}, t), P(\mu_{v_i}, t), P(comm_vole_{e_{ij}}, t), P(\epsilon_{v_i}, t), P(d_{v_i}, t)) \quad (2.4) \\
& \quad \text{and architectural components} \\
& Arch \left\{ T(U, F), \Omega, S_P, P(r_i, t), P(ch_{ij}, t), P(freq_{p_k}, t) \mid r_i \in U, ch_{ij} \in F, p_k \in S_P \right\} \\
& g_\zeta(App, Arc, M, S, R, t) \leq h_\zeta, \quad \zeta = 1, \dots, N \text{ are satisfied}
\end{aligned}$$

where M represents the mapping function, S denotes the scheduling function, R is the routing policy. The g_ζ functions depend on various stochastic processes characterizing both the application (*e.g.*, packet generation/consumption) and architecture (*e.g.*, router service time), while h_ζ are the design constraints that need to be satisfied. In other words, the goal here is to determine a particular NoC architecture (*e.g.*, specific topology, channel bandwidth, buffer sizes, *etc.*), a mapping, scheduling and/or a routing function that satisfy the imposed design constraints which can represent performance (*e.g.*, bandwidth, throughput, node-to-node latencies), power/energy consumption, and/or reliability metrics. For instance, one can consider to minimize the buffer size $l(r_i, ch_{ij})$ of all channels ch_{ij} connected to router $r_i \in U$ across an NoC architecture such that, for some information about the packet generation ($P(\lambda_{v_i}, t)$) and consumption ($P(\mu_{v_i}, t)$),

²Here optimal refers to the minimum number of cores that execute the code to meet the performance and deadline constraints with minimum energy consumption. However, this definition can be relaxed or changed depending on the chosen cost and constraint functions.

the deadline constraints (h_ζ) for each task are met. In this case, the functions g_ζ would represent the sum between the time it takes to process a set of tasks on which task $v_i \in V_{App}$ depends on and the node-to-node latency between the cores on which the tasks are mapped. Note that the role of time-dependent PDFs is to capture the existing correlations at the application level; this is even stronger emphasized by the mapping operation, packet scheduling, routing and the generated network traffic on some particular performance metrics like node-to-node latencies.

Chapter 3

Multi-core Traffic Characterization

3.1 Rationale for Multi-core Traffic Monitoring

Accurate *traffic modeling* has profound implications, not only on performance evaluation, but also in various optimization aspects, such as scheduling, dynamic power management, chip temperature regulation, *etc.* Running full-blown applications requires either complete products (too late to allow any major changes) or system-level models (too slow). At the same time, real traces cannot be directly used in any optimization loop as they are strongly constrained by the characteristics of the architecture. Consequently, relying on a more accurate traffic characterization for NoC performance analysis and optimization is a fundamental task in *network-based design*.

Most current traffic models rely on exponential inter-arrival times and stationary stochastic processes [87]. Simply speaking, under this assumption, the probability of large time intervals between two consecutive packet arrivals is decaying exponentially to zero. The stationary assumption implies that the distribution and implicitly the moments of the stochastic process are time independent.

Nevertheless, these models cannot capture many of the traffic characteristics (*e.g.*, time mono-fractality¹) observed via power laws of header inter-arrival times shown in Figure 3.1.a. Figure 3.1.b shows that inter-arrival times are better fitted by a stable distribution with long tail (*e.g.*, Levy distribution, Cauchy distribution, Holtsmark distribution) rather than exponential or Gaussian laws. Moreover, due to the heterogeneous traffic sources and continuous changes in the pool of applications running on a multicore platform, at various time intervals, the traffic variation *cannot* be characterized by a single fractal dimension² (*i.e.*, mono-fractal behavior), but rather by a *series* of interwoven time scales defined by several such fractal dimensions (*i.e.*, multi-fractal behavior).

Towards this end, since the traffic variability at multiple scales is caused by both architectural (*e.g.*, routing protocol, routing service policy, buffering resources) and application/user (*e.g.*, variable packet sizes, packet injection periodicity, scheduling dependency) features, and to better

¹Time mono-fractality refers to a stochastic process $x(t)$ characterized by a single fractal dimension H and for which all its q -th order moments satisfy a time power law of the following form $K_q \approx t^{qH}$ [54][102][125][148].

²The fractal dimension quantifies how the size of a fractal object changes under the magnification operation [102].

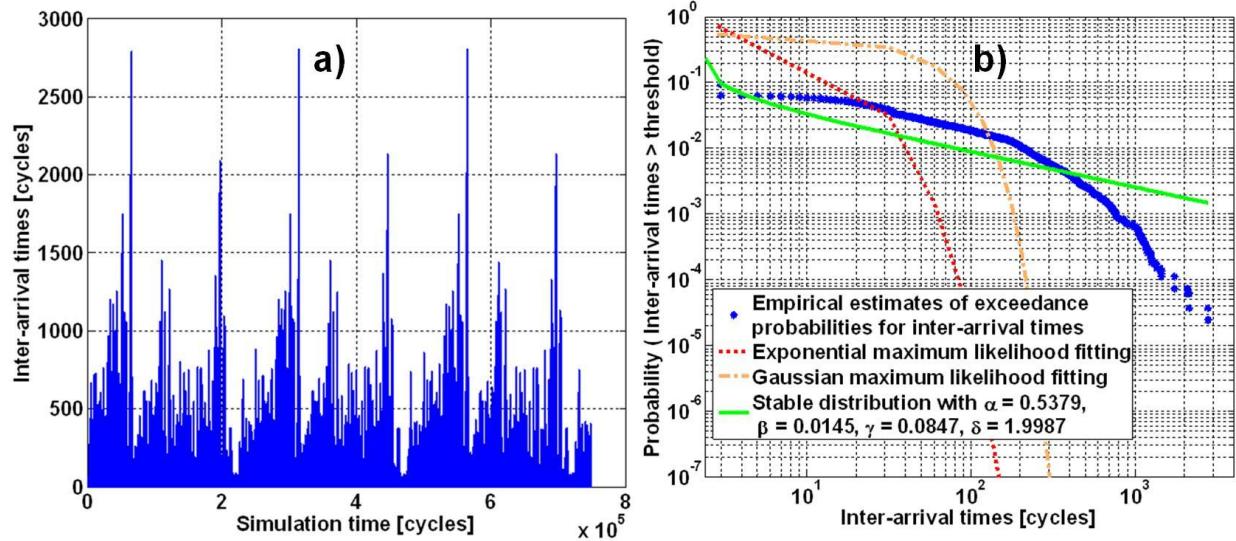


Figure 3.1: a) The header inter-arrival times are measured at the West input buffer of node (1,2) on a 4×4 mesh NoC (see Figure 3.2) with input and output buffers of 3 and 1 slot, using XY wormhole routing with packets consisting of 15 flits and running a multi-threaded online transaction processing (OLTP TPC-C v3.0, IBM DB2 v8 ESE & Oracle 10g Enterprise Database Server) application. b) Probability of inter-arrival times to exceed a given threshold does not follow an exponential or gaussian distribution, but rather is better fitted by a stable distribution. In addition, one can observe that there exists a change in slope of the tail distribution which is a sign that the inter-arrival time stochastic process exhibits multiple fractal exponents.

capture the heterogeneity and non-stationarity of NoC traffic, we propose a statistical physics inspired model which allows us to characterize the packet arrival process via a histogram of fractal dimensions. As we will show throughout this thesis, this does not only allow for better modeling of network traffic traces, but also paves the way of defining new dynamic optimization methodologies.

3.2 Related Work and Major Contributions

Traffic analysis is a fundamental problem in many research domains ranging from vehicular and transportation systems [70], to communication [95][120] and biological networks [137]. Although many approaches have been proposed for traffic modeling (both at microscopic and macroscopic levels [96]), many of the fundamental traffic characteristics (*e.g.*, non-stationary, self-similarity, *etc.*) still cannot be fully captured by state-of-the-art techniques [120].

Since reviewing the literature available on traffic modeling in an exhaustive manner is beyond the scope of this thesis section, in what follows, we focus primarily on traffic models proposed for communication networks. Most traffic models based on queueing theory [52][56][66][74][117] rely on the assumption of having a Poisson packet arrival process (*i.e.*, the time intervals between two consecutive packet arrivals are assumed to be exponentially distributed) and a random service time that is independent of the arrival process. An alternative perspective to queueing theory models is represented by deterministic bounds obtained via network calculus concepts [11][129]. None of

these approaches to performance evaluation of interconnection networks considers non-stationary effects in traffic behavior³.

Moreover, as shown already by several studies, the traditional traffic models based on queueing theory [87] *cannot* capture the fractal nature of traffic observed in Internet [59][120][121] or NoCs [138][145][154]. Simply speaking, these measurement-driven studies argue that, in many instances, the packet arrival/departure rate described by random variable $x(t)$ is *self-similar*⁴ in nature and just *a single exponent* (*i.e.*, the Hurst parameter H) is enough to capture the traffic characteristics. From a theoretical stand point, this implies that the moments⁵ $K_q(t)$ of the stochastic process $x(t)$ scale as $K_q(t) = \int_0^\infty x^q P(x, t) dx \propto t^{qH}$, where $P(x, t)$ is the probability that $x(t)$ attains value x at time t .

However, in this chapter, we claim that due to the heterogeneous traffic sources and continuous changes in the pool of applications running on a multicore platform, the traffic variation *cannot* be characterized by a single exponent H , but rather by a series of interwoven time scales defined by several such exponents at various time intervals. Mathematically speaking, this implies that the moments associated with various traffic processes (*e.g.*, packet arrival/departure process, packet waiting times in buffers, and so on) need to obey scaling laws of the following form $K_q(t) = \int_0^\infty x^q P(x, t) dx \propto t^{f(q)}$ where $f(q)$ is a nonlinear function of the moment index q . In fact, the existence of this nonlinearity of the exponent $f(q)$ as a function of q implies that the stochastic process is multi-fractal.

One way to elucidate the need for a multi-fractal approach (instead of a mono-fractal one which can be characterized by second-order statistics, like mean and variance) is to analyze the higher order statistics of traffic processes (*e.g.*, packet inter-arrival times). Higher order statistics (HOS) provide us with mathematical tools (*e.g.*, third/fourth-order moments) to detect deviations from the Gaussian distribution and quantify the nonlinearity exhibited by various stochastic processes [105]. For instance, due to intrinsic symmetry, the third-order central moment (*i.e.*, $E[(X - E[X])^3]$) of a Gaussian process is zero. By contrast, if the third order moment is not zero than the stochastic process cannot be assumed to be Gaussian distributed.

For example, the third-order moment of the packet inter-arrival times process at the North input buffer of node located at (1, 2) in Figure 3.3 is 32.1823. We note that, as shown in Figure 3.3, these packet inter-arrival times are obtained by running a multimedia application on a 4×4 mesh NoC (see Figure 3.2) with input and output buffers of size 10 slots, under wormhole XY routing (packets consist of 5 flits). This shows that second-order statistics (which would correspond to a

³A stochastic process is called non-stationary if its mean, variance, and all higher order moments are *not* invariant under arbitrary time shifts.

⁴Generally speaking, a stochastic process exhibits a self-similar behavior if the time series of its realization over some time scale (*e.g.*, microseconds, milliseconds) and its re-scaled realization via a single scale parameter over another time scale (*e.g.*, minutes, hours) look similar. Self-similarity was first addressed in a series of papers by Kolmogorov [88] and later applied to study random processes in the absence of asymptotic independence [102].

⁵The moment index q denotes the order of the expectation operator (*i.e.*, the q -th moment); this is a quantitative measure of the probability distribution [87]. For instance, the second moment of a stochastic process measures how spread a set of points are. The third central moment (*i.e.*, $E[(X - E[X])^3]$) quantifies the asymmetry of the distribution.

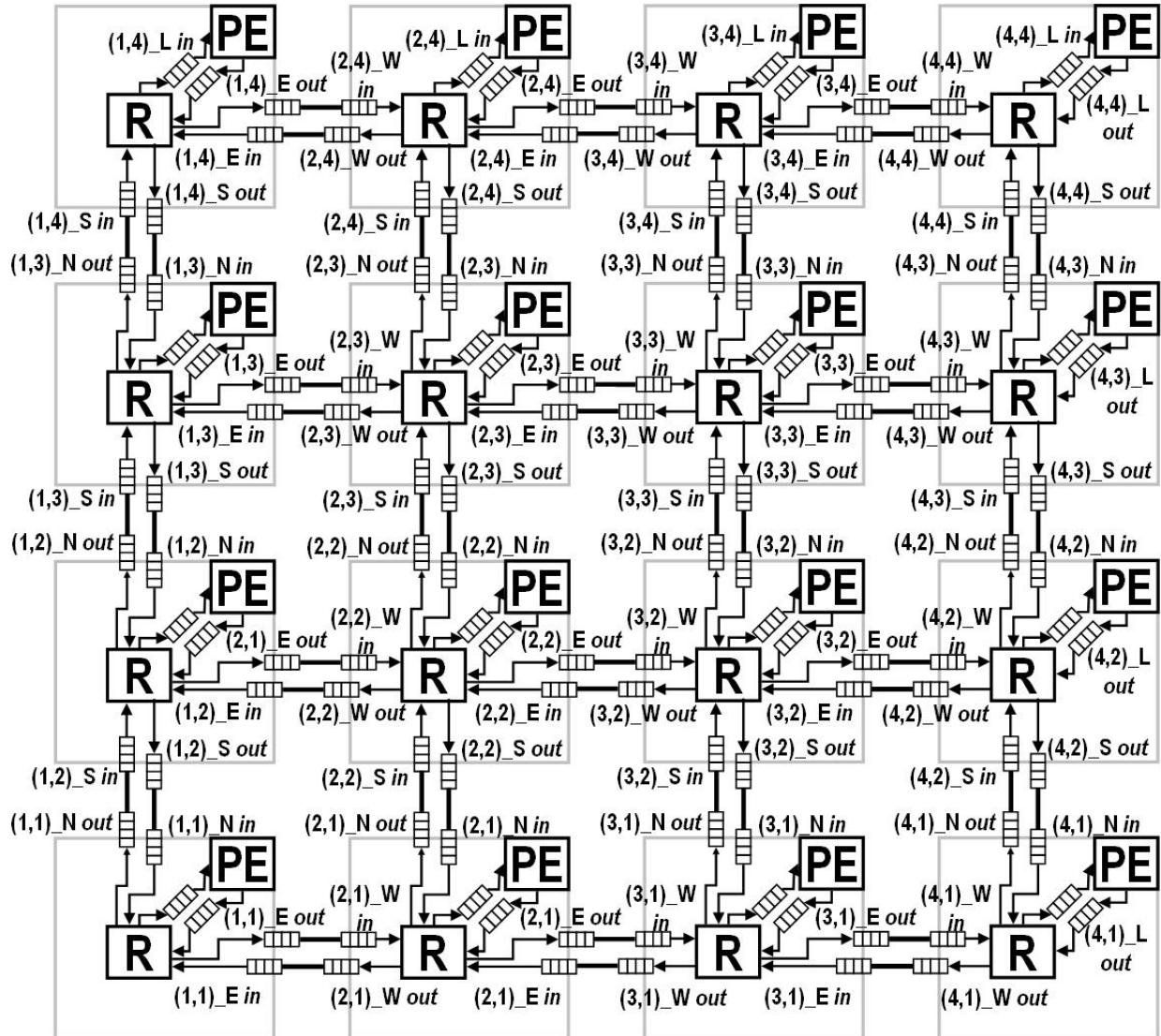


Figure 3.2: Diagram of a 4×4 mesh NoC where the symbols L, N, E, W, and S stand for the local, north, east, west, and south router connections, and *in* and *out* suffixes represent the input and output directions.

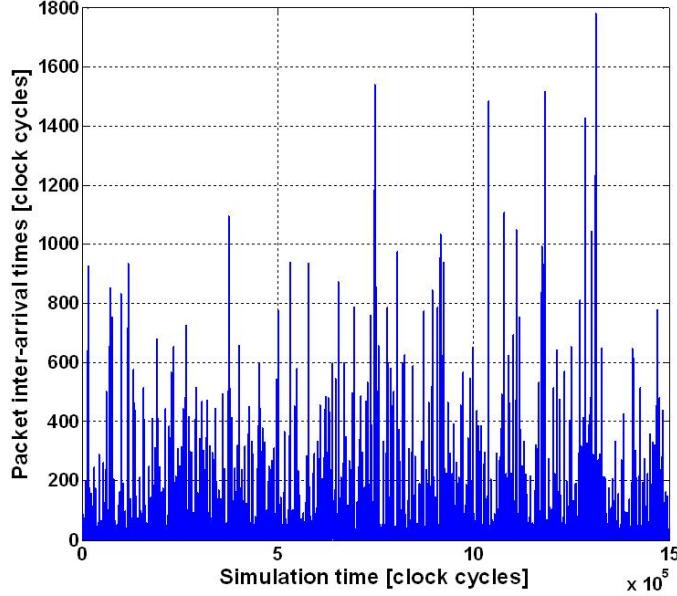


Figure 3.3: Packet inter-arrival times at the North input buffer of node (1, 2) on a 4×4 mesh NoC (see Figure 3.2) with input and output buffers of size 10 slots, under wormhole XY routing (packets consist of 5 flits) and running a multimedia application.

mono-fractal analysis) *cannot* fully capture the complexity of the network traffic. Consequently, the present work aims at capturing such complex characteristics of traffic in a new multi-fractal framework based on HOS concepts.

One natural question at this point is related to the possible sources for this multi-fractal behavior of the NoC traffic. Although this is a hard question to answer, we can speculate that one source comes from the intrinsic characteristics of the applications such as variability in input processing (*e.g.*, differences in context based decoding of high definition video frames) and patterns observed in the number of the injected packets and their timing intervals. Also, based on user preferences and other dynamical properties, certain applications get replaced by others with specific quality and timing constraints. This can also imply changes in the source-to-destination traffic patterns. So the traffic variability at multiple scales is caused by both architectural (*e.g.*, routing protocol, routing service policy, buffering resources) and application/user (*e.g.*, variable packet sizes, packet injection periodicity, scheduling dependency) characteristics. We propose a statistical physics inspired model which allows us to characterize the packet arrivals in the network as a multi-fractal and non-stationary process evolving both in space and time; this is described next.

3.3 A Statistical Physics Inspired Model for NoC Traffic

We model the packets transmission in the NoC via a random graph (RG) (see Figures 3.4 and 3.5), where the nodes and the edges represent the buffers in the NoC architecture and the packets exchanged between these buffers, respectively [23]; also, the number of arriving and departing

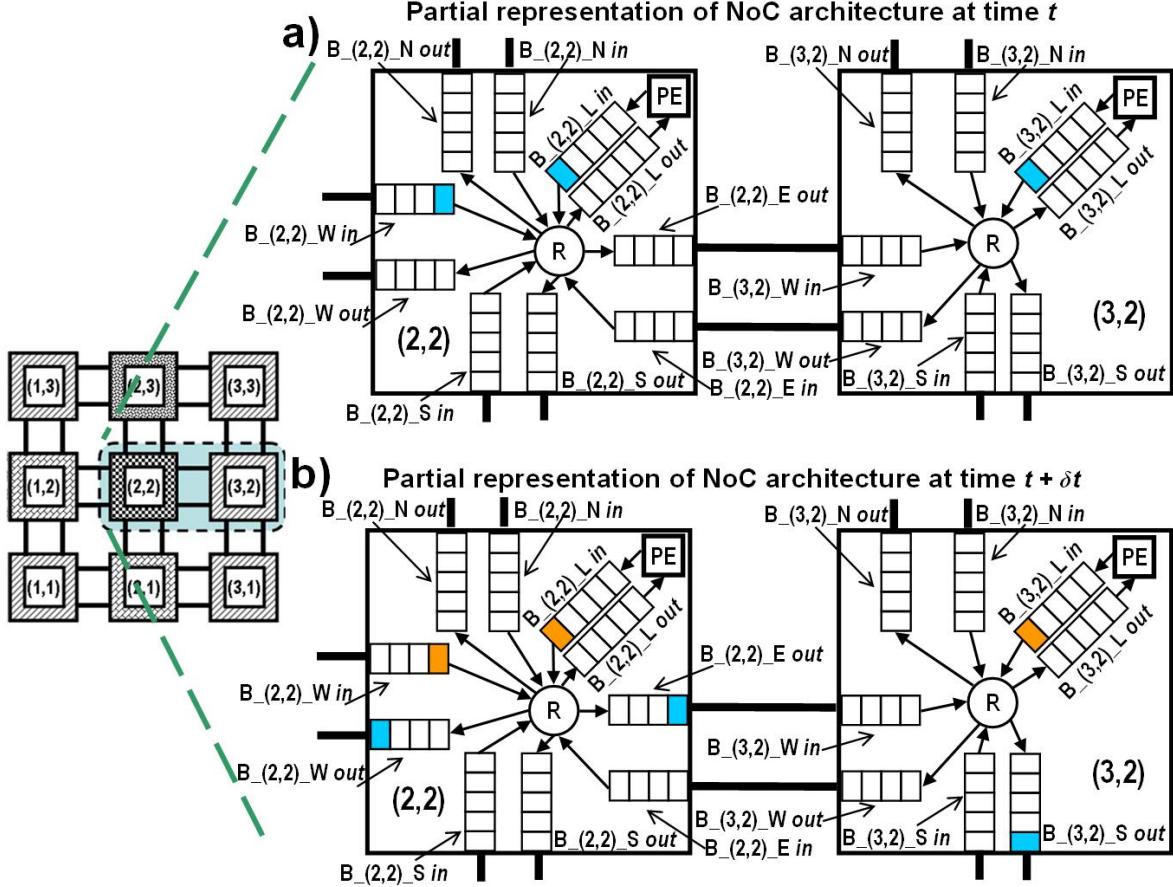


Figure 3.4: Snapshot of the NoC architecture at time t (a) and time $t + \delta t$ (b). Each NoC node consists of a processing element (PE), a router (R) and several input and output buffers corresponding to the number of its neighbors. The highlighted slots of the buffers represent the buffer occupancy at a particular time. For instance, between t and $t + \delta t$, one packet is injected from the PE at (2,2) to the local input buffer, one packet previously in the input buffer of the PE at (2,2) is routed towards node (1,2) and a new packet is received at West input buffer from node (1,2).

packets represent the *IN* and *OUT* degree of each node, respectively. In this model, the traffic in the network is represented via a dynamic RG where packets follow their source-to-destination paths and thus changing continuously the distribution of the RG nodes characterized by a certain *IN* and *OUT* degree. It is important to note that this RG is simply an abstraction meant to capture the dynamics of the packets flow in the NoC rather than the real network with a physical (fixed) topology.

To be more concrete, Figure 3.5 shows how the traffic passing through two NoC nodes can be represented with this RG-based model. More precisely, each buffer in the NoC (namely, the *L*, *N*, *E*, *W*, and *S* labels standing for Local, North, East, West and South directions and with *in* and *out* suffixes representing the input and output channel buffers) is represented as a node in the RG. The numbers on the arrows connecting the RG nodes show the actual *IN* and *OUT* degree of each node as a result of various network transactions taking place at a particular time t .

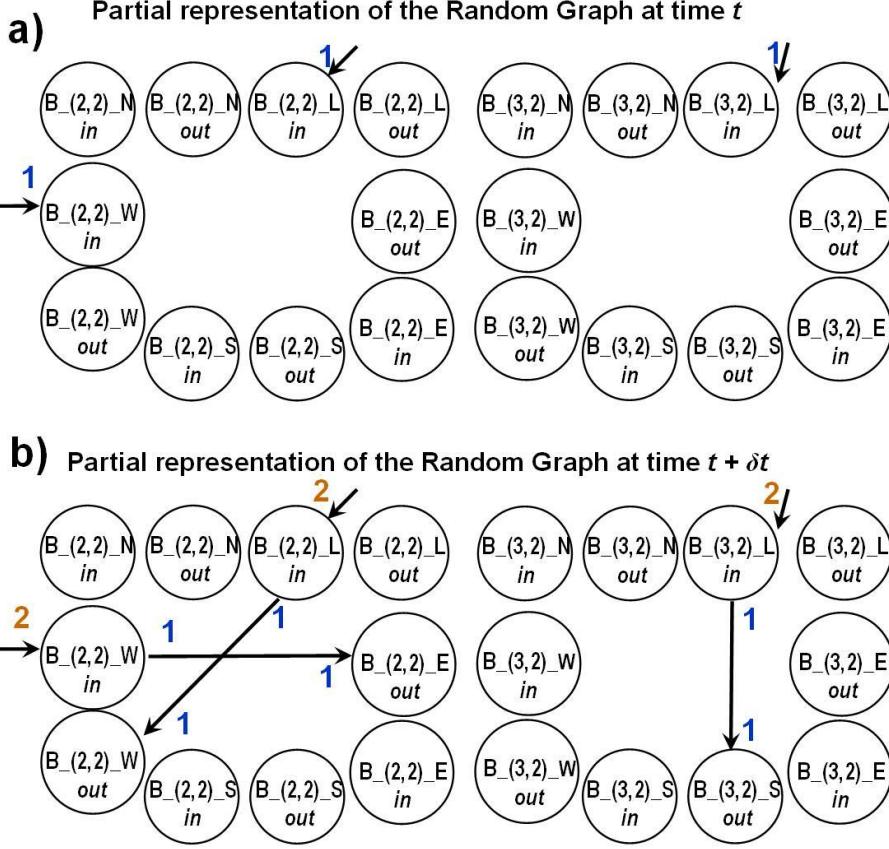


Figure 3.5: Snapshot of two NoC nodes as a random graph (RG) at time t (a) and $t + \delta t$ (b). Each node in the RG represents a buffer in the NoC (e.g., $B_{(2,2)}\text{.L.in}$ denotes the local input buffer between the PE and the router at (2,2) mesh coordinates). The numbers on the arrows show the *IN* and *OUT* degree of RG nodes. As shown above, the *IN* and *OUT* degree of the $B_{(2,2)}\text{.W.in}$ RG node is 1 and 0, respectively, at time t , and 2 and 1, respectively, at time $t + \delta t$. The unconnected RG nodes, having zero *IN* and *OUT* degree, correspond to buffers not included in the communication process yet. The difference between the *IN* and *OUT* degree at a particular time t shows the buffer occupancy.

For instance, $B_{(2,2)}\text{.E.out}$ in Figure 3.5 denotes the East output buffer located at the mesh location (2,2) which has an *IN* and *OUT* degree of 0 and 0, respectively, at time t , and an *IN* and *OUT* degree of 1 and 0, respectively, at time $t + \delta t$. Obviously, the dynamics of the RG changes over time and this is driven by the actual communication volume of packets exchanged among the NoC nodes. In addition, one can foresee that a packet transmission from the local input buffer to the East output buffer at node (2,2) would take a random amount of time dictated by the router service policy, size of the previously sent packets, and the time stamp of the current transaction. This randomly distributed waiting time can be studied via a stochastic analysis of the *IN* and *OUT* degree of the RG nodes. In addition, note that there is an intrinsic connection between the arrival time to an NoC buffer and the *IN* degree of the corresponding node in the RG. Consequently, in what follows, we describe an analytical approach for quantifying the dynamics of nodes *IN* degree.

One way to model the *IN* degree dynamics of a node characterized by η and θ fitness functions is to write a rate equation determining the average number of nodes $N(i, t|\eta, \theta)$ having degree i by time t :

$$\begin{aligned} \frac{\partial N(i, t|\eta, \theta)}{\partial t} = & \frac{\eta(t)}{H(t)}(i-1)N(i-1, t|\eta, \theta) - \frac{\eta(t)}{H(t)}iN(i, t|\eta, \theta) + \\ & + \frac{\theta(t)}{K(t)}(i+1)N(i+1, t|\eta, \theta) - \frac{\theta(t)}{K(t)}iN(i, t|\eta, \theta) \end{aligned} \quad (3.1)$$

where the first term represents the addition of one edge to a node having *IN* degree $i-1$ and so contributing to an increase in $N(i, t|\eta, \theta)$, the second term represents the addition of one edge to one of the nodes having *IN* degree i which leads to a subtraction of $N(i, t|\eta, \theta)$, the third term denotes a loss of one edge from a node having $i+1$ *IN* degree which increases the average number of nodes $N(i, t|\eta, \theta)$ by one and the last term represents the corresponding loss of one node from $N(i, t|\eta, \theta)$ due to an edge deletion. The function $H(t)$ and $K(t)$ are introduced for normalization purposes and are given by the following relations:

$$H(t) = \sum_{i, \eta(t)} i\eta(t)N(i, t|\eta, \theta) \quad K(t) = \sum_{i, \theta(t)} i\theta(t)N(i, t|\eta, \theta) \quad (3.2)$$

An alternative approach is to investigate the time-dependent probability distribution of any node in the RG, characterized by fitness functions η and θ , to have an *IN* degree i , $P(i, t|\eta, \theta)$, defined as follows:

$$\begin{aligned} P(i, t + \delta t|\eta, \theta) = & \frac{\eta(i-1)\delta t}{M(t)t^\beta}P(i-1, t|\eta, \theta) + \frac{\theta(i+1)\delta t}{Z(t)t^\beta}P(i+1, t|\eta, \theta) + \\ & + \left[1 - \left(\frac{\eta i}{M(t)} + \frac{\theta i}{Z(t)} \right) \frac{\delta t}{t^\beta} \right] P(i, t|\eta, \theta) \end{aligned} \quad (3.3)$$

In other words, Eq. 3.3 states that the dynamics of the probability $P(i, t|\eta, \theta)$ of a node in the RG, characterized by fitness functions η and θ , to have an *IN* degree i at time t , is proportional to the weighted contribution of three possible outcomes: The first term on the right hand side of Eq. 3.3 reflects the increase in the *IN* degree which is taking place with a time dependent rate $[\eta i \delta t] / [M(t)t^\beta]$, the second term models the loss in the *IN* degree due to congestion in the network, while the third term stands for the “no change” state in the *IN* degree.

Note that, in order to account for various correlations in the arrival times (and implicitly in the *IN* degree) of any RG node, we model the probability of a new arrival event as being proportional to the previous cumulative number of arrival events weighted by a random variable η which is meant to capture the utilization of this RG node at any point in time. Note that depending on the HOS characteristics exhibited by the packet arrival process, the edge attachment probability can have either a linear expression as in Eq. 3.3 or a nonlinear one such as i^α .

The role of the fitness η is not only to weight the contribution of past packet arrivals, but also to discriminate between buffers across NoC which are exercised differently by the traffic patterns in the network.

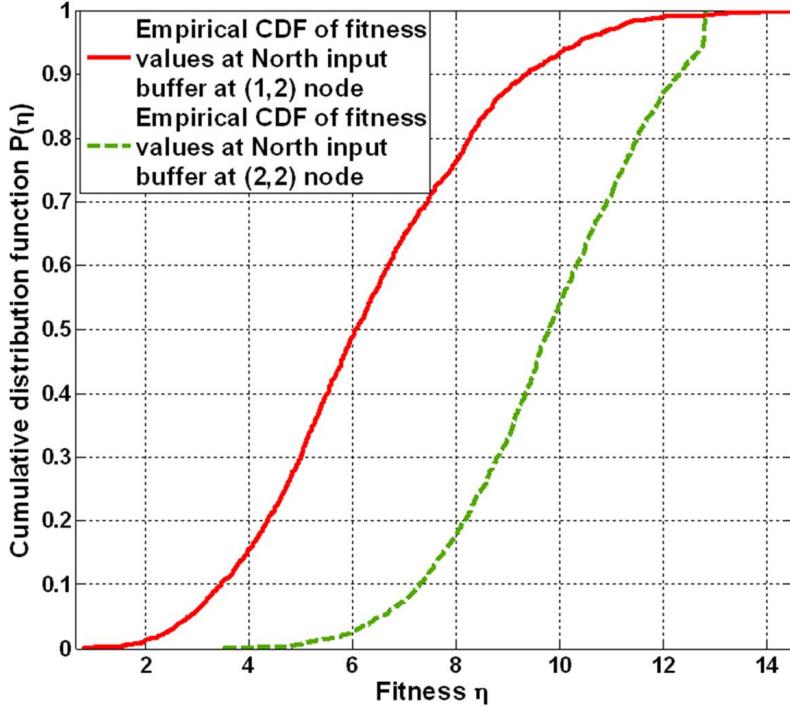


Figure 3.6: Empirical cumulative distribution function of the fitness values obtained from two *IN* degree time series corresponding to the North input buffers of the nodes located at (1, 2) and (2, 2), respectively, for a 4×4 mesh NoC running an MPEG4 decoder application (see Figure 4.1) and an average packet injection rate per node of 0.029 packets per cycle.

To better illustrate the heterogeneity among NoC buffers captured via fitness values, we report in Figure 3.6 the empirical cumulative distribution function of the fitness associated with the *IN* degree time series of North input buffers at nodes (1, 2) and (2, 2), respectively, in a 4×4 mesh NoC with input and output buffers of 20 and 10 slots, respectively, employing an XY wormhole routing protocol (7 flits per packet) and running an MPEG4 decoder application (see Figure 4.1) and an average packet injection rate per node of 0.029 packets per cycle. One can notice not only the difference in the slope of the two CDFs, but also the variation (heterogeneity) in the support of fitness values (*i.e.*, the support of the fitness η for the North input buffer at (2,2) ranges between 3.8 and 12.5 abstract units, while for the North input buffer at (1,2) ranges between 0.2 and 14.5).

Coming back to Eq. 3.3, we also introduce a parameter $0 \leq \beta \leq 1$ to capture the memory effects of the packet arrival process. More precisely, for $\beta = 0$, we obtain a dynamic equation characterizing a short-range dependence packet arrival process. Finally, the $M(t)$ and $Z(t)$ are slowly varying functions playing a normalization role and are given by the following relations:

$$M(t)t^\beta = \sum_{i,\eta(t)} i\eta(t)N(i,t|\eta,\theta) \quad Z(t)t^\beta = \sum_{i,\eta(t),\theta(t)} i\theta(t)N(i,t|\eta,\theta) \quad (3.4)$$

To ease the analysis of Eq. 3.3, we make use of the finite difference method approximations [128] to express the two terms, (*i.e.*, $(i+1)P(i+1,t|\eta,\theta)$ and $(i-1)P(i-1,t|\eta,\theta)$, respectively).

This leads to the following continuum version of the previous master equation:

$$\frac{\partial \left[t^\beta P(i, t | \eta, \theta) \right]}{\partial t} = \left[\frac{\eta}{M} + \frac{\theta}{Z} \right] \frac{\partial^2 [iP(i, t | \eta, \theta)]}{\partial i^2} + \left[\frac{\theta}{Z} - \frac{\eta}{M} \right] \frac{\partial [iP(i, t | \eta, \theta)]}{\partial i} \quad (3.5)$$

Note that for $\beta = 0$, Eq. 3.5 reduces to a diffusion-like equation with load dependent coefficients [87]. A simplified version of Eq. 3.5 describing the dynamics of the probability that a stochastic process attains value x at time t via two fractal exponents, one for the power law inter-events and the other one for the magnitude increments in x variable, was proposed in [26]. Nevertheless, the real-time applications exhibit a much more complex behavior due to the multi-user preferences and interactions with system resources, hence our detailed analysis here.

In summary, we analyze the RG dynamics as a function of application characteristics (*e.g.*, packet injection rate, number of already acquired packets at a particular buffer, traffic patterns) and architecture features (*e.g.*, routing protocol, buffering resources) which are encompassed via fitness functions.

Chapter 4

Implications of Traffic Fractality

4.1 Analytical Implications of the Master Equation to Multi-core Traffic Modeling

In order to analyze the time-dependent (non-stationary) behavior of the higher order moments associated with packet arrival processes (*i.e.*, the *IN* degree of an RG node), we introduce the following expression of the high-order moments $K_q(t|\eta, \theta)$ [54][81][102] associated with the probability distribution function $P(i, t|\eta, \theta)$ as follows:

$$K_q(t|\eta, \theta) = \sum_{i=0}^{\infty} i^q P(i, t|\eta, \theta) \approx \int_0^{\infty} i^q P(i, t|\eta, \theta) di \quad (4.1)$$

where q represents the moment order. More precisely, by introducing the higher order moments in Eq. 4.1, we establish a connection between the dynamic behavior of the moments and the intrinsic parameters of the NoC traffic like the fitness function η associated with the packet arrival process to a certain buffer in the network. This connection is essential not only because it shows how the nonlinearity observed in the higher order moments behavior is affected by changes in the NoC traffic traces, but also because these parameters (*e.g.*, memory parameter β , fitness η) play a significant role in various performance metrics (*e.g.*, buffer overflow probability, node-to-node latency exceedance probability) that can guide the NoC optimization.

Multiplying by i_q to the left hand side and summing up between 0 and ∞ in Eq. 3.5 leads to the following differential equation for the moments:

$$\begin{aligned} \frac{\partial \left[t^\beta K_q(t|\eta, \theta) \right]}{\partial t} &= - \left[\frac{\eta}{M} + \frac{\theta}{Z} \right] q \frac{\partial [q K_q(t|\eta, \theta)]}{\partial q} + \\ &+ \left[\left(\frac{\eta}{M} + \frac{\theta}{Z} \right) q^2 - \left(\frac{\theta}{Z} - \frac{\eta}{M} \right) q \right] K_q(t|\eta, \theta) \end{aligned} \quad (4.2)$$

For $\beta = 1$, the moments $K_q(t|\eta, \theta)$ of distribution $P(i, t|\eta, \theta)$ scale as a power law form: $K_q(t) \approx g(q)t^{f(q)}$, where $g(q)$ and $f(q)$ represent the weighting coefficients and the mass ex-

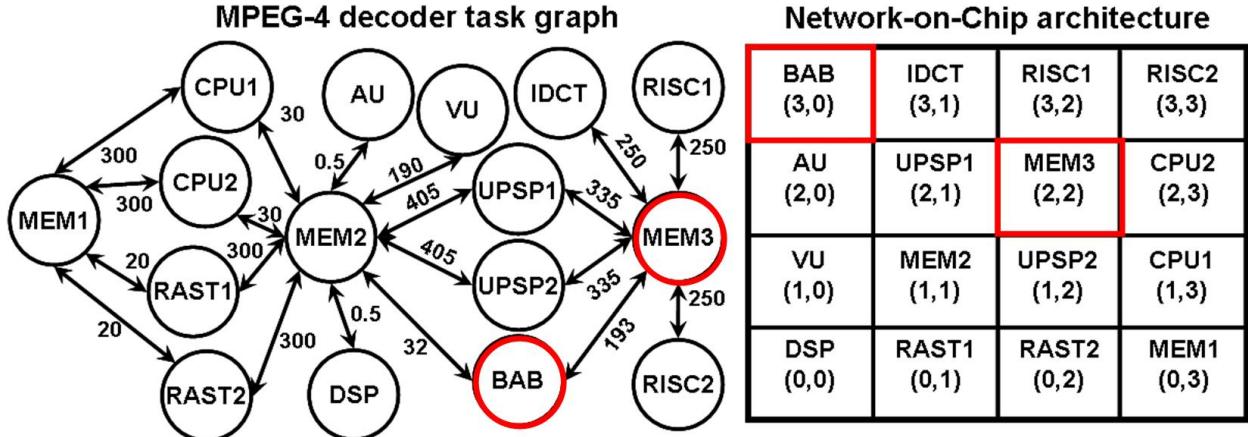


Figure 4.1: MPEG4 decoder task graph [133] mapped onto a 4×4 mesh NoC architecture.

ponents, respectively, as a function of the moment order q . If the mass exponent function $f(q)$ is nonlinear in the moment order q , then the stochastic process, characterized by $P(i, t|\eta, \theta)$, is called multi-fractal [102]. Instead, if $f(q)$ is linear in the moment order q , then the stochastic process is called mono-fractal.

Starting from the power law solution of Eq. 4.2, one can obtain the multi-fractal spectrum $h(\alpha)$ via the Legendre transform [102]:

$$f(q) = \alpha(q) - h(\alpha) \quad \alpha = \frac{\partial f(q)}{\partial q} \quad (4.3)$$

where α denotes the *fractal dimension* associated with the packet arrival process to a specific buffer. Generally speaking, Eq. 4.3 shows that a multi-fractal process can be seen as a statistical ensemble of many (individual) fractal dimensions which are intrinsically related to the deep characteristics of application and architecture.

In summary, we propose a new analytical model that captures the multi-fractal characteristics and the non-stationary behavior of the NoC traffic. In the next sections of this chapter, we validate our findings and show how this formalism can be employed for NoC design and optimization.

4.2 Critical Phenomena in Multi-core Systems

To better understand the impact of multi-fractal behavior on the performance of NoCs, we first present in Table 4.1 the deviation of the header flit inter-arrival times from an exponential distribution as a function of packet injection rate in an MPEG4 decoder application. This application was mapped on a 4×4 mesh NoC as shown in Figure 4.1, with the input and output channel buffers of 20 and 10 slots, respectively, while employing an XY wormhole routing protocol (7 flits/packet). During the mapping process of the MPEG4 decoder application on a 4×4 mesh NoC, we tried to keep the highly communicating task nodes close to each other. This is the reason behind placing the two highly utilized memory blocks (*i.e.*, MEM2 and MEM3) on the tiles in the middle of the

Average Packet Injection Rate (λ)	Percentage of Exponentially Distributed Inter-Arrival Times	Network Throughput	Throughput Increase
0.005 pkts/cycles	83%	0.083	1(base line)
0.006 pkts/cycles	66%	0.103	1.25
0.029 pkts/cycles	33%	0.465	5.60
0.032 pkts/cycles	25%	0.517	6.23
0.034 pkts/cycles	16%	0.534	6.43
0.035pkts/cycles	10%	0.54	6.51

Table 4.1: NoC performance as a function of packet injection rate for a 4×4 mesh with input and output buffers of 20 and 10 slots, using XY wormhole routing and running an MPEG4 decoder (see Figure 4.1 for MPEG4 task graph details). Higher packet injection rates cause a decrease in the percentage of exponentially distributed inter-arrival times.

NoC.

The results in Table 4.1 are obtained by averaging over tens of simulation configurations consisting of 10^7 clock cycles each and while using various random seeds for each run. To derive the statistical results of the second column, each header inter-arrival time corresponding to an input buffer was tested via the Kolmogorov-Smirnov (KS) test. The KS test quantifies the minimum distance between an empirical distribution function of some experimental samples and a postulated cumulative distribution function [157]. In this setup, we perform a null hypothesis that the empirical header inter-arrival times come from an exponential distribution. We accept the null hypothesis at 95% significance level saying that the samples are well approximated by exponential distribution.

As can be seen from the second column of the Table 4.1, as the average packet injection rate per node increases from 0.005 to 0.035 packets/cycle, the percentage of header flit inter-arrival time processes that can be fitted by an exponential distribution decreases from 83% to 10%, respectively. For instance, for a 0.006 (packets/cycle) average packet injection rate per node, the percentage of exponentially distributed header inter-arrival times is 66% (*i.e.*, there are 32 true Poisson processes). In other words, this shows that as the packet injection increases, the NoC traffic deviates significantly from the Poisson assumption by having fewer exponentially distributed packet inter-arrival times.

More importantly, one can see that for the case when only 25% of the header inter-arrival times can be regarded as being exponentially distributed, the network throughput is more than *6 times larger* compared to the case when the exponential behavior dominates (*i.e.*, at 0.005 packets/cycle). This implies that, as the 4×4 mesh NoC gets closer to its criticality regime, a phase transition phenomena takes place and so the Poisson type of distribution *cannot* be used anymore to estimate performance or optimize the network! Instead, our proposed approach which considers network traffic via multiple fractal dimensions is the way to go about performance analysis. At the same time, a multi-fractal approach overcomes the drawbacks of mono-fractal models which require that traffic traces need to be infinitely long.

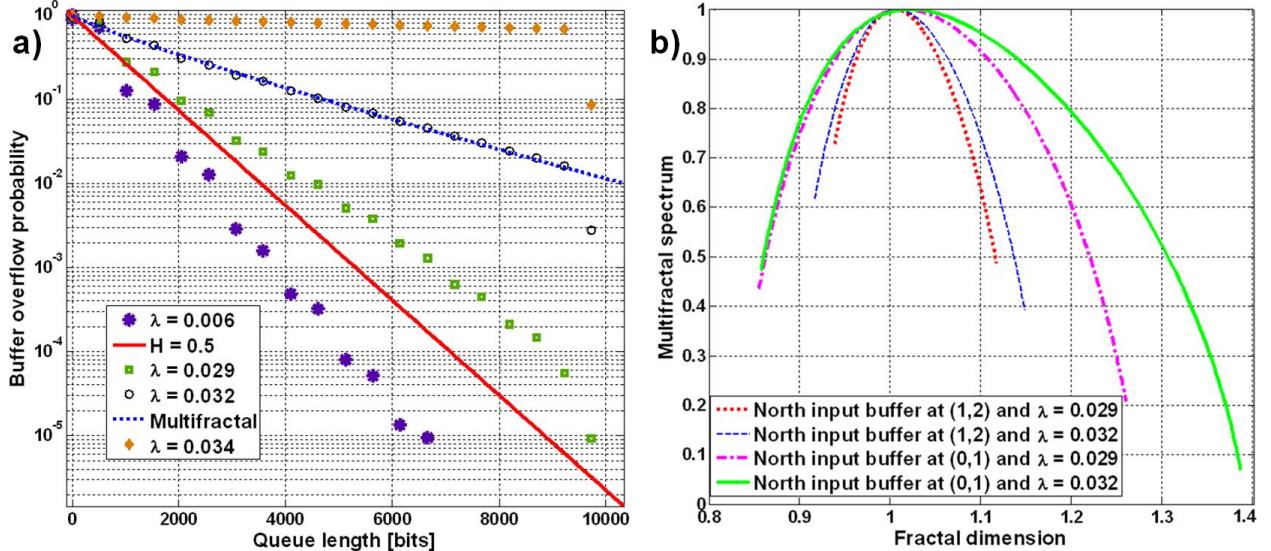


Figure 4.2: a) The buffer overflow probability as a function of packet injection rate for the input buffer to a memory block of an MPEG4 decoder mapped on a 4×4 mesh NoC with finite buffers (input and output channel buffer sizes of 20 and 10 slots, respectively) and XY routing scheme. With higher packet injection rates, the buffer overflow probability exhibits a nonlinear behavior which can be captured only by using a multi-fractal approach. b) The multi-fractal spectrum as a function of the fractal dimension of the header inter-arrival times at the North input buffer at nodes (1, 2) and (0, 1), respectively (see Figure 3.2 for the diagram of a 4×4 mesh NoC).

4.3 Implications of NoC Traffic Multi-fractality on Buffer Sizing

In Figure 4.2.a, we show how this multiscale approach can help us estimate the performance of a concrete NoC platform. More precisely, we analyze the buffer overflow probability for the input buffer to a memory block (*i.e.*, node (2, 2) in Figure 4.1.a which receives requests from the binary alpha block (BAB), inverse discrete cosine transform (IDCT) and several RISC PEs in an MPEG4 application. Performing this kind of analysis is important since an overflow at this buffer can result in performance degradation of the video decompression because the BAB module is in charge of decoding the shape via context-based arithmetic decoding [133].

With higher packet injection rates, the buffer overflow probability exhibits a nonlinear behavior which can only be accounted for by considering multiple fractal exponents. For instance, the probability of having a queue length of 6000 bits is approximately 1000 times higher when the average packet injection rate per node increases from 0.006 to 0.032 (see Figure 4.2.a). Note that the solid line denotes the overflow probability under the Poisson assumption, while the dotted line stands for the multi-fractal case.

To prove the existence of multi-fractal behavior of traffic *just before congestion*, Figure 4.2.b shows the multi-fractal spectrum of the arrival times to two North input buffers of nodes at (1, 2) and (0, 1), respectively, and for two packet injection rates (*i.e.*, 0.029 and 0.032 packets/cycles). As one can notice, with the increase of packet injection rate, the two time series of packet arrival

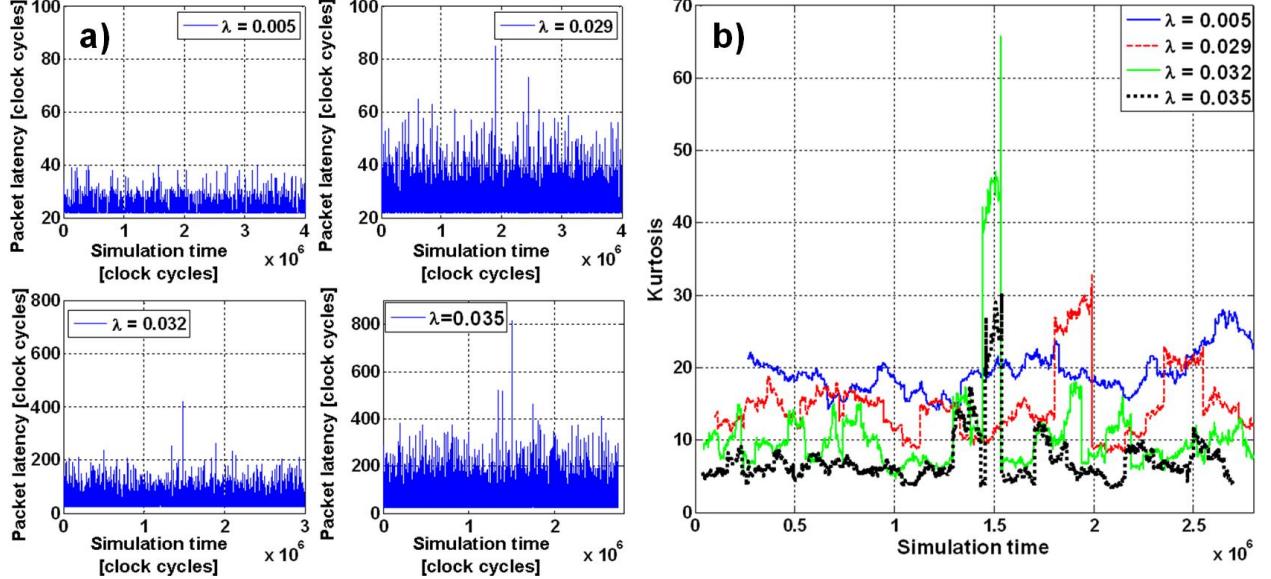


Figure 4.3: a) Time series of packet latencies between a source node at (3,0) and a destination node at (2,2) for four packet injection rates. b) Kurtosis of the packet latency as a function of the simulation time and packet injection rate. The exhibited variation in the kurtosis trend represents not only the presence of a non-Gaussian behavior, but also a non-stationary signature.

process at the input buffers exhibit a much broader spectrum of fractal dimensions. For instance, for the packet arrival process at North input buffer at (0, 1) node the maximum fractal dimension increases from 1.25 to 1.39 while the minimum fractal dimension remains almost the same. This also shows the lack of accuracy related to the mono-fractal modeling.

4.4 Implications of Traffic Multi-fractality on Estimating the Node-to-Node Latency

Next, we investigate the impact of operating the NoC architecture close to criticality and its implications on *packet latency* estimation. More precisely, we plot in Figure 4.3.a the variation of packet latency between the BAB module at location (3, 0) (see the mapping in Figure 4.1.a) and the MEM3 memory block at location (2, 2) as a function of simulation time for four packet injection rates (λ), namely 0.005, 0.029, 0.032, and 0.035 packets per cycle. As one can observe, higher packet injection rates cause the magnitude of the packet latency to increase quite significantly (see plots in Figure 4.3.a).

To stress the importance of higher order statistics for this type of variation, we report in Figure 4.3.b the kurtosis of the packet latency between nodes (3, 0) and (2, 2) as a function of the simulation time, for four different packet injection rates. The *kurtosis* (*i.e.*, the ratio between the fourth order moment and the standard deviation of a probability distribution) captures the frequency of rare events¹. Note that, for each packet latency (which has a corresponding arrival time stamp at the

¹A rare event in the context of node-to-node latency refers to a very large latency, sometimes few orders of

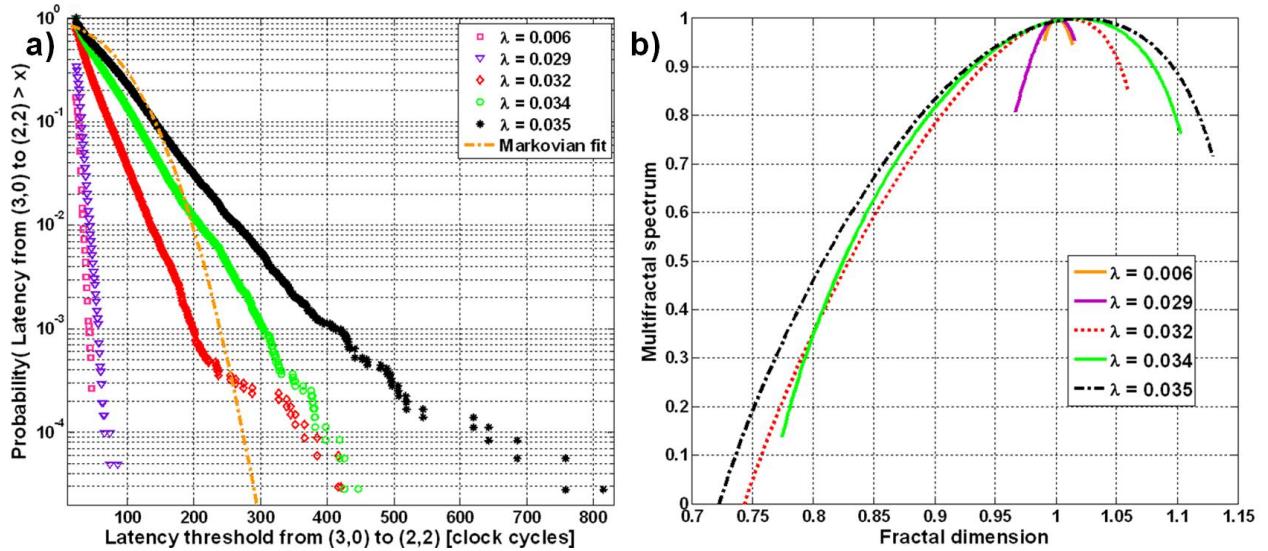


Figure 4.4: a) Probability of packet latency between nodes (3,0) and (2,2) to exceed a certain threshold for an MPEG4 decoder mapped on a 4×4 mesh NoC (see Figure 3.2) with finite buffers (input and output buffer sizes of 20 and 10 slots, respectively), wormhole switching (7 flits per packet), XY routing scheme and several packet injection rates. b) The multifractal spectrum as a function of fractal dimension of the packet latencies between source at (3,0) and destination at (2,2) and several packet injection rates.

destination on the x-axis of Figure 4.3.b, we compute the kurtosis as a function of simulation time over a sliding window consisting of 500 forward and backward packet latency values. Increasing the packet injection rate causes the kurtosis to exhibit a more spiky behavior which, again, is a sign of non-stationary behavior. Similar non-stationary trends have also been observed for first three moments (*i.e.*, mean, variance, skewness). The significant (positive) deviations in Figure 4.3.b also show that the distribution of packet latency *does not* follow Gaussian laws, but rather it follows a power law instead. This also illustrates that the node-to-node latency exhibits a nonlinear behavior which needs to be accounted for by any packet scheduling approach.

To be more concrete, in Figure 4.4.a we report the probability of the latency between the BAB module at (3,0) and the MEM3 memory block at (2,2) to exceed a certain threshold for four packet injection rates. This probability can be interpreted as the probability of missing a deadline when scheduling since there are several computational modules waiting for packets to arrive. To better emphasize the importance of a multi-fractal approach in NoC run-time optimization, we also plot the packet latency Markovian exceedance probability (see dash-dot line). Quantifying accurately this probability is important since higher delays encountered between the BAB module and the memory block can result in poor quality decompression and identification of shapes in the video frames.

By way of contrast, while the Markovian curve predicts a zero probability for the latency magnitude larger, compared to the mean value. Adopting a Gaussian or Poisson type of mathematical framework to predict such extreme events can result in poor predictions.

between locations (3, 0) and (2, 2) to exceed 300 clock cycles, the experimental results show a 0.0002 exceedance probability for $\lambda = 0.032$ packets per cycle, a 0.0011 exceedance probability for $\lambda = 0.034$ packets per cycle, and a 0.0055 exceedance probability for $\lambda = 0.035$ packets per cycle. Consequently, we claim that *any* Markovian model based on Poisson-type of distribution can lead to performance results that severely underestimate the probability of exceeding a certain packet latency or the probability of missing a given deadline. Markovian approaches rely most frequently on one or two parameters (mean and variance) for estimating the probability of a certain event. Considering only the mean and variance implies to approximate the probability of an extreme event by a Gaussian law which *underestimates* the chance of rare events [100]. An alternative to this approach consists of increasing the significance of large events by considering higher order moments (*e.g.*, kurtosis).

To investigate the existence of multi-fractal behavior in packet latency time series between the BAB module at (3, 0) and the memory block at (2, 2), we report in Figure 4.4.b their multi-fractal spectrum as a function of the fractal dimension for several packet injection rates. From these results, one can observe that with increasing packet injection rates, the multi-fractal spectrum does not only shift towards right (*i.e.*, a region corresponding to more correlated traffic), but also gains with respect to the width of fractal dimensions (see the x-axis); this is a clear sign of the departure from mono-fractal behavior². When increasing the packet injection rate, the dominant fractal dimension corresponding to the maximum value attained by the multi-fractal spectrum shifts from 1 (for $\lambda = 0.032$), to 1.017 (for $\lambda = 0.034$) and to 1.021 (for $\lambda = 0.035$).

The existence of nonlinearity exhibited in Figure 4.4.a and the multi-fractal behavior observed in packet latency time series shown in Figure 4.4.b imply that any real-time scheduling strategy should not be based on a linear time theory, but instead consider nonlinear approaches (*e.g.*, nonlinear feedback control strategies, stochastic optimization of higher order moments). Of note, a similar multi-fractal behavior has been observed in [27] for the data request-reply latencies of various SPEC-2006 applications running on CMP-based NoC architectures.

4.5 Where Does the Traffic Model Fit in the NoC Design and Optimization Flow?

The multi-fractal nature of the NoC traffic has profound implications in various optimization domains (*e.g.*, packet scheduling, dynamic power management, chip temperature regulation, *etc.*). The newly proposed approach (shown schematically in Figure 4.5) does not necessarily complicate the network optimization task, but rather it motivates the need to shift the focus from linear systems theory and control based on mean values (*e.g.*, average buffer/link utilization, average PE utilization, deadline miss ratio) to control and optimization approaches based on higher order statistics (*e.g.*, third and forth order moment statistics). Failure to do so can result in highly non-

²The shift towards right can be seen by observing the maximum value attained by the multi-fractal spectrum which corresponds to the dominant fractal dimension of the packet latencies.

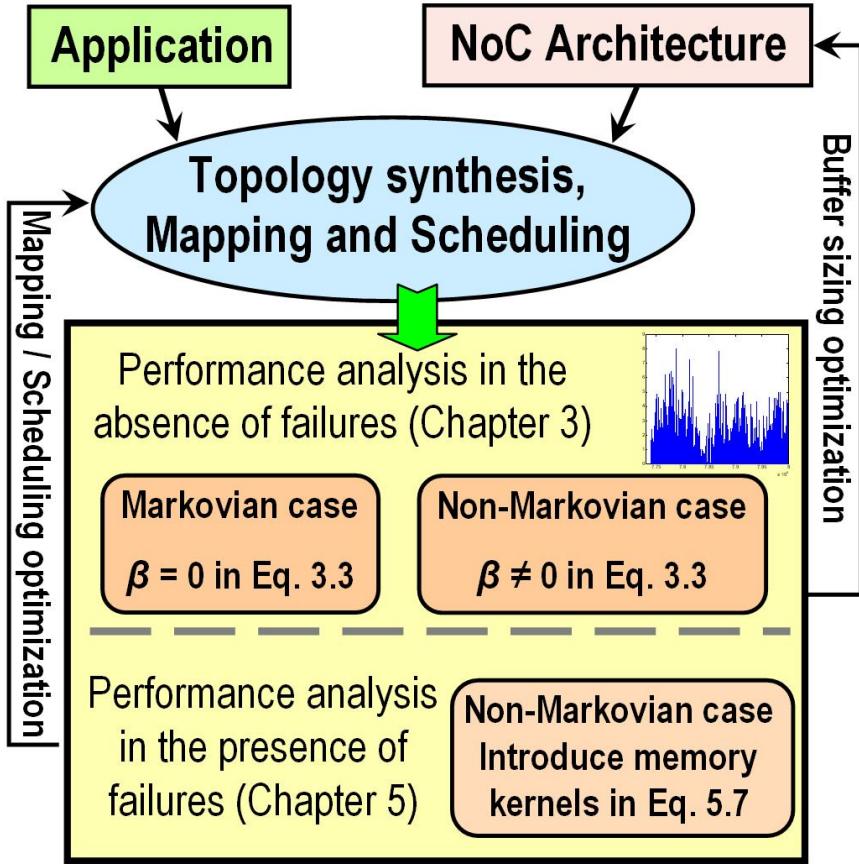


Figure 4.5: Proposed traffic analysis methodology and its implications at application and architectural levels. Starting from the application and architecture characteristics, we build a dynamic probabilistic model of the network traffic which can be used for solving various problems such as topology synthesis, mapping, scheduling or routing.

optimal solutions in resource allocation and management (*e.g.*, large buffer size, slow convergence of the synthesized controllers for dynamic power management, *etc.*) or even unwanted behaviors in the network response (*e.g.*, sporadic and uncontrolled congestion).

Chapter 5

Modeling and Analysis of Fault-tolerant NoC Communication Protocols

5.1 Biologically Inspired Communication Protocols for NoCs: Stochastic Communication

So far we were concerned with dynamic processes taking place on networked architectures in the absence of hardware failures. However, shrinking transistor dimensions, smaller interconnect features and higher operating frequencies lead to a higher sensitivity of deep-submicron (DSM) circuits to neutron and alpha radiation, significantly higher soft-error rates, and an increasing number of timing violations [142]. These types of failures are impossible to characterize using deterministic approaches and, thus, probabilistic metrics (*e.g.*, hitting probabilities) are needed to quantify the critical design objectives, such as fault-tolerance, performance and power. Traditional acknowledgement/request protocols are not adequate in such an error prone environment.

To reduce the prohibitive cost of design and verification in the DSM domain and to provide a high system-level fault-tolerance, a new communication paradigm, called *stochastic communication* was proposed [57]. Under stochastic communication, data from various computational modules are encapsulated into packets at the IP level and then it is probabilistically disseminated from sources to destinations¹. As shown in Figure 5.1, each tile keeps a list of packets (Send_Buffer) that have to be sent to the output buffers. The packets received during the last communication round and the new packets generated by the IP are added to this buffer. However, if a packet is already stored, a duplicate packet will not be inserted. At each intermediate node, the protocol evolves as follows: If a packet is successfully received, it is first checked for information integrity via a cyclic redundancy check (CRC) (see Figure 5.1). If the packet is neither corrupted, nor a duplicate, then the logic

¹Note that we assume that the destination node has a local buffer capable to store the received out-of-order packets and hardware logic to re-order them based on the information contained in the packets header.

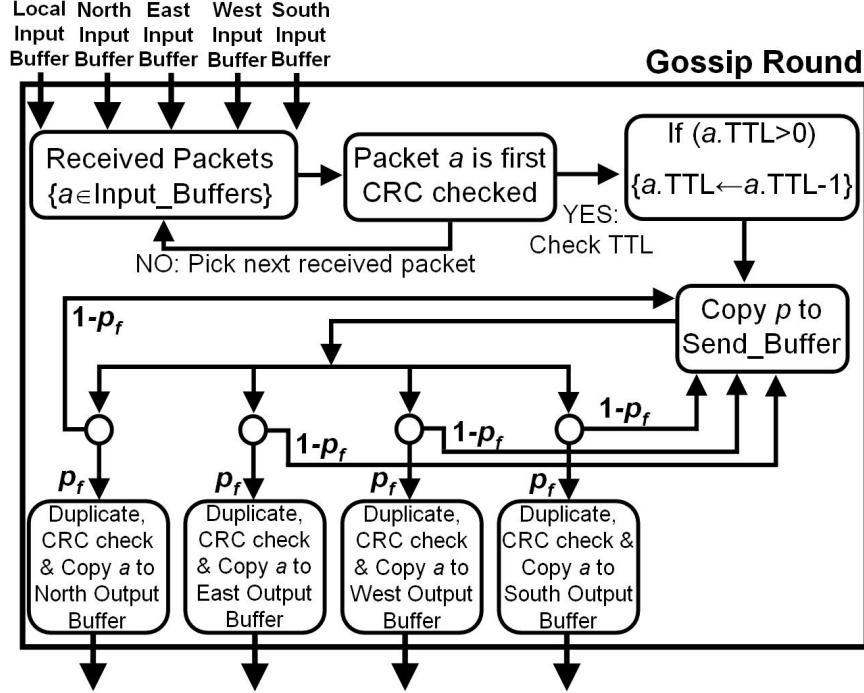


Figure 5.1: Stochastic communication consists of a Gossip Round executed concurrently by all nodes. A packet received at the input ports is first CRC checked for information integrity. If the packet is successfully received, then its TTL parameter is decreased and the packet is copied to the Send_Buffer. According to the forwarding probability (p_f) the packet is duplicated, CRC checked (to prevent errors occurred during routing decision) and copied to the output buffers.

circuitry of the router decrements the time-to-live (TTL) parameter and probabilistically sends it to a subset of its neighbors. Note that for simplicity reasons, we assume that the TTL value is large enough (*e.g.*, comparable to the order of the network diameter) to ensure that packet dissemination does not terminate at some intermediate node along the path from source to destination.

5.2 Related Work and Major Contributions

When dealing with communication errors, one of the primary properties of a communication protocol is the degree of fault-tolerance. Most of the times, the degree of fault-tolerance of a communication protocol has been quantified through *node coverage*² metric. Traditionally, analytical models used in computer networks or data bases to quantify the fault-tolerance of various communication protocols relied on mathematical models (*e.g.*, rate equation, master equation) developed in epidemics [10][49] and rumor spreading [122] theory.

Unlike epidemics and rumor spreading theory, which assume a homogeneous structure between spreader³ and receiver individuals, we constructed a *master equation* quantifying the probability of

²The definition of *node coverage* refers to the number of nodes in the network that received at least one copy of the disseminated message.

³In epidemics, a spreader individual denotes an agent that is infected with a virus and spreads it continuously or

having a particular number of spreader⁴, ignorant⁵, and stifler⁶ nodes at a particular time t while taking into account the network topology[24]. Although beneficial for determining the percentage of network nodes aware of a piece of information, this approach does not offer information about concrete performance metrics (*e.g.*, latency) that can guide the overall NoC design process from both architectural (*e.g.*, determine the best communication protocol as a function of technology) and application (*e.g.*, find the best clustering, partitioning and mapping of an incoming application [104]) perspectives.

Starting from these ideas, we construct an analytical approach for computing the *mean hitting time* between any two nodes in a network that operates under the stochastic communication protocol. In contrast to most mathematical approaches studying the gossip style protocols which typically quantify the number of nodes aware of some piece of information, the hitting time analysis allows us to estimate how much time is needed to reach, from any given source, a particular destination node in an arbitrary network. Consequently, evaluating the mean hitting time in a network where communication happens stochastically is analogous to evaluating the latency in a network where communication happens in a completely deterministic manner. This makes it clear how important is to have available such a performance metric in order to design and evaluate a stochastic communication protocol.

The concept of *hitting time*, or the *first passage time* as it is known in statistical physics, has been mostly related to the problem of reaching an absorbing state (*e.g.*, quantifying the existence of steady-states for Rayleigh model of a gas, extinction of a population in prey-predator type of problems [111]). The time characteristics of absorbing/extinction processes have been investigated through the hitting time density function (*i.e.*, the probability that the time at which a particle reaches an absorbing state x_1 is within the interval $(t, t + \delta t)$, given that the particle starts at position x_0 and time $t = t_0$).

In the context of random walks theory, the hitting time was originally developed in connection with electrical networks [43][55]. Since its inception, the theory of random walks attracted significant attention due to its potential for solving a variety of problems in diverse application domains such as the design of distributed computation [40][43][60], estimation of the complexity of distributed algorithms [41], search in peer-to-peer networks [63], estimation of Web size [14]. While the evolution of a simple random walk on graphs is extensively studied in the literature, only recently there has been some interest in the study of multiple, yet finite number of random walks [6][48].

Nevertheless, many natural phenomena [10] and human-driven processes [15] cannot be modeled using a single or only a finite number of random walks. Thus, in what follows, we model the

over a finite interval of time to any other individual with equal probability. Similarly, in rumor spreading processes, a spreader refers to an agent that is aware of the rumor and chooses to spread it to all other individuals with equal probability. Note that, in general, topological restrictions are not considered.

⁴The spreader represents any node that is aware of the message and decides to disseminate it.

⁵The ignorant defines a node that is not included in the communication area yet.

⁶The stifler node refers to the case in which a node aware of the message chooses to cease its dissemination (either the link is faulty or the node is dropping the packet).

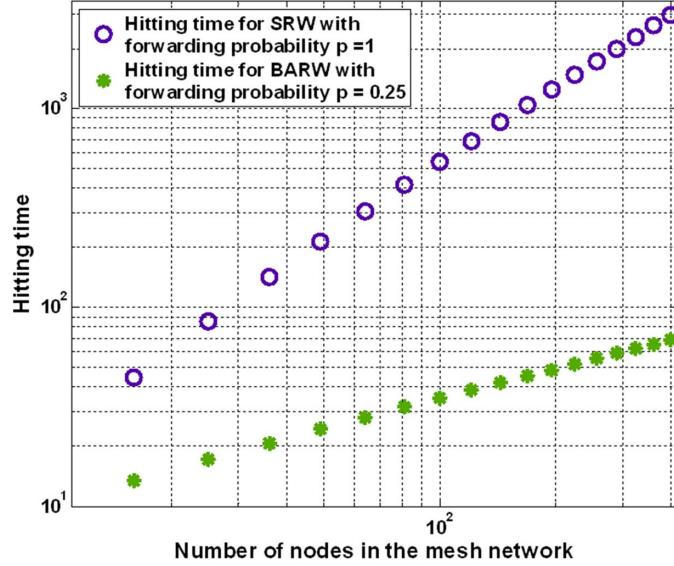


Figure 5.2: Mean hitting time (*i.e.*, time elapsed while propagating a packet from source $(1, 1)$ to destination (N, N)) comparison between SRW and BARW approaches for various mesh sizes $(2 \times 2, \dots, 20 \times 20)$. The results are conservative as we consider a best case scenario for SRW (*i.e.*, the packet is sent to a randomly chosen neighbor in each round) and an average case scenario for BARW (*i.e.*, packets are forwarded with 0.25 probability).

stochastic communication process as a *collection of random walks* taking place on finite graphs and investigate their evolution (dynamics) both in space and time.

To illustrate the impact of the mesh topology on the dynamics of the branching process, as well as the nature of the epidemic style spreading, we report on a log-log scale the average hitting time as a function of the network size (see Figure 5.2), when the initiating source and destination nodes are located at nodes $(1, 1)$ and (N, N) , respectively, under the following two scenarios. A *single random walk* (SRW) where at each location on a mesh a virus residing at the current node infects with probability 1 a single random neighbor without considering the neighbors initial state, and an epidemic style spreading where at each location on a mesh, the virus at the current node infects an arbitrary subset of neighbors; this is similar to information spreading in stochastic communication. In the context of stochastic communication, a packet can be corrupted during link transmission or simply dropped by a faulty node; this epidemic style spreading is seen as a branching and annihilating random walk (BARW) process [42][150].

5.3 A Branching and Annihilating Random Walks Approach to Modeling Fault-Tolerant Communication Protocols

Next, we focus on estimating the performance (*e.g.*, *hitting time*⁷) of stochastic communication routing seen as a collection of dynamic processes taking place on a network in the presence of errors.

⁷The hitting time represents the time it takes for a packet to propagate from a source to a destination node.

More precisely, we model the packet probabilistic dissemination as a *branching random walk*, while the reverse process, in which packets are corrupted or overwritten, as an *annihilated random walk* [116]. In contrast to a single random walk where, at any given time, a message is sent randomly only to a neighboring node [58][55][132], the stochastic communication allows for *multiple random walks* starting at each node; this provides a much higher dissemination speed and robustness to the communication protocol compared to a simple random walk. Consequently, one can say that stochastic communication *generalizes* the simple random walk model, both in time and space, and so it needs a new and more general theory to explain it.

Adopting a statistical physics approach, we model the communication events (*e.g.*, packet duplication, packet dropping) during a probabilistic dissemination initiated at node $(i, j)_S$ and taking place on an $N \times N$ mesh network by associating to each node (i, j) a stochastic process $\{O_{ij}(t) | 1 \leq i, j \leq N, t \geq 0\}$ which reflects the number of received copies at any time t . The packet dissemination across the network can be described by the following interactions:

a) Packet duplication: In a short time interval δt , a node (i, j) can duplicate a packet according to the relation:

$$Pr\{O_{ij}(t + \delta t) = k + 1 | O_{ij}(t) = k\} = \lambda_{ij}k\delta t + O(\delta t) \quad (5.1)$$

where λ_{ij} is the *packet duplication rate* for each node (i, j) , k is the number of received copies, and $O(\delta t)$ is a negligible term. The duplication process starts only if the node (i, j) received at least one copy (*i.e.*, $k > 0$), otherwise the duplication probability is zero. The packet duplication rate λ_{ij} is proportional with the packet forwarding probability of the dissemination algorithm as shown in Figure 5.3.

b) Packet successful transmission: A packet duplicated at node (i, j) can be successfully sent to its neighbor at $(i, j - 1)$ (thus increasing its O_{ij-1}) as follows:

$$\begin{aligned} Pr\{(O_{ij}, O_{ij-1})(t + \delta t) = (k - 1, n + 1) | (O_{ij}, O_{ij-1})(t) = (k, n)\} = \alpha_{ij}^W k \left(1 - \frac{n}{B_{ij-1}}\right) \delta t + \\ + O(\delta t) \end{aligned} \quad (5.2)$$

where α_{ij}^W is the *link successful transmission rate* from node (i, j) to its West neighbor $(i, j - 1)$ (*i.e.*, the router at (i, j) sends a copy to $(i, j - 1)$ node and the packet is successfully received), k and n represent the number of received copies at $(i, j - 1)$ and (i, j) nodes at time t , and B_{ij-1} is the size of the buffer at $(i, j - 1)$ node. The transition probability $Pr\{(O_{ij}, O_{ij-1})(t + \delta t) | (O_{ij}, O_{ij-1})(t)\}$ in Eq. 5.2 is strictly positive only if the sender node (i, j) has received at least one copy (*i.e.*, $k > 0$). Moreover, this probability is meant to account for all errors that occur during the link transmission due to induced noise (*e.g.*, ground bounce, inductive and capacitive crosstalk, IR drop, thermal noise, *etc.*) which are detected by the CRC module and the buffer overflows. Indeed, excessive packet duplication can cause buffer overflows; this is captured by the $(1 - O_{ij-1}/B_{ij-1})$ term, with B_{ij-1} being the buffer size at node $(i, j - 1)$. If the buffer at node O_{ij-1} is empty, then this term has no effect on the transition probability. If the number of received copies O_{ij-1} increases, then this

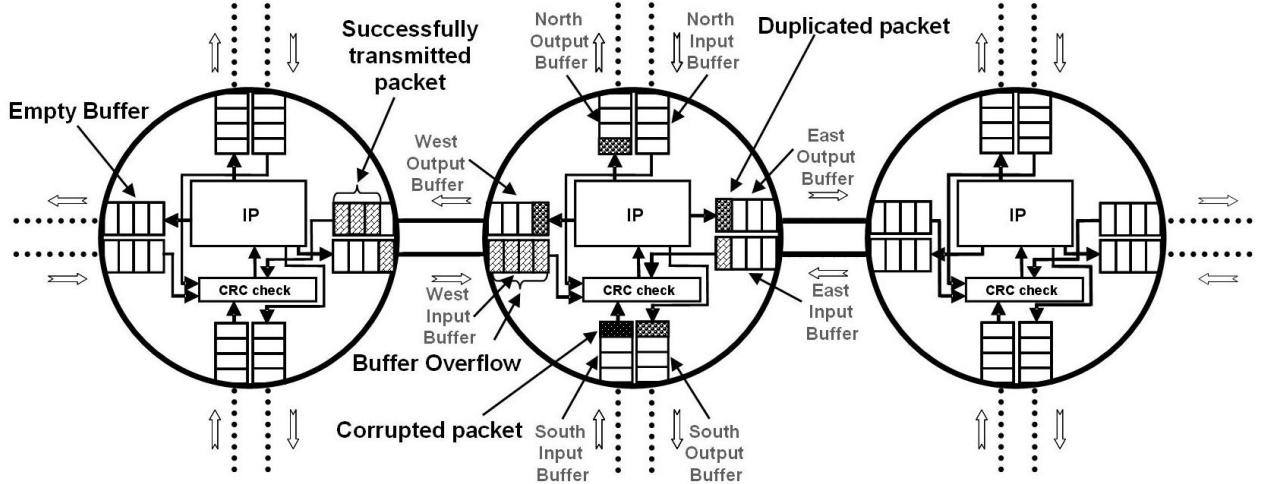


Figure 5.3: Possible events (*e.g.*, packet duplication, packet successful transmission, buffer overflow, and packet corruption) between two neighboring nodes under the stochastic communication protocol.

probability decreases. Similarly, we can describe the transmission events from (i, j) to the North (O_{i-1j}), East (O_{ij+1}), and South (O_{i+1j}) neighbors.

c) Packet corruption while routing: The probability that a node corrupts a (received) packet during the routing decision, and then causes it to be discarded, is:

$$Pr \{O_{ij} (t + \delta t) = k | O_{ij} (t) = k + 1\} = \mu_{ij} (k + 1) \delta t + O(\delta t) \quad (5.3)$$

where μ_{ij} denotes the *packet corruption rate* at node-level. This transition is activated with rate μ_{ij} only if the node (i, j) has already received a positive number of packets ($k > 0$). This accounts for potential errors in computation at node-level. For instance, erratic bit phenomena have been reported in [3] to occur in SRAMs due to random noise which affects the minimum voltage at which the memory cell remains functional. This implies that packets received successfully can be occasionally corrupted (*e.g.*, bits in the header flit can be flipped) inside the memory cells or during the routing decisions before actually being routed. More importantly, it is predicted that these effects will only get worse in future DSM technologies [3][47].

d) No interaction: In case of no communication events, the corresponding transition probability is given by:

$$\begin{aligned} Pr \{O_{ij} (t + \delta t) = k | O_{ij} (t) = k\} &= 1 - O(\delta t) - \lambda_{ij} k \delta t - \mu_{ij} \lambda_{ij} k \delta t \\ &- \left[\alpha_{ij}^N \left(1 - \frac{m_{i-1j}}{B_{i-1j}} \right) + \alpha_{ij}^E \left(1 - \frac{m_{ij+1}}{B_{ij+1}} \right) + \alpha_{ij}^W \left(1 - \frac{m_{ij-1}}{B_{ij-1}} \right) + \alpha_{ij}^S \left(1 - \frac{m_{i+1j}}{B_{i+1j}} \right) \right] k \delta t \end{aligned} \quad (5.4)$$

where m_{i-1j} , m_{ij-1} , m_{ij+1} , and $m_{i+1,j}$ denote the number of received copies by the neighboring nodes of node (i, j) .

Having already characterized these individual interactions, the packet diffusion over the entire

network can be seen now as a *collection* of random walks, where the occurrence probability of any transition in an interval $(t, t + \delta t)$ relies on the number of packets received at time t , at each node. The nodes evolution (or dynamics) can be described via a master equation of a multivariate probability distribution function:

$$P(o_{11}, \dots, o_{ij}, \dots, o_{NN}; t) = \Pr\{O_{11}(t) = o_{11}, \dots, O_{ij}(t) = o_{ij}, \dots, O_{NN}(t) = o_{NN} \mid O_{11}(0) = m_{11}, \dots, O_{ij}(0) = m_{ij}, \dots, O_{NN}(0) = m_{NN}\} \quad (5.5)$$

which shows that the stochastic process $O_{ij}(t)$ associated to node (i, j) received o_{ij} packets ($O_{ij}(t) = o_{ij}$) by time t .

The evolution of the probability distribution in Eq. 5.5 is given by the following differential equation:

$$\begin{aligned} \frac{P(\dots, o_{ij}, \dots; t)}{dt} &= \sum_{i,j=1}^N \{\lambda_{ij}(o_{ij} - 1) P(\dots, o_{ij} - 1, \dots; t) + \mu_{ij}(o_{ij} + 1) P(\dots, o_{ij} + 1, \dots; t) + \\ &+ \alpha_{ij}^N(o_{ij} + 1) \left(1 - \frac{o_{i-1j} - 1}{B_{i-1j}}\right) P(\dots, o_{i-1j} - 1, o_{ij} + 1, \dots; t) + \alpha_{ij}^E(o_{ij} + 1) \left(1 - \frac{o_{ij+1} - 1}{B_{ij+1}}\right) \bullet \\ &P(\dots, o_{ij} + 1, o_{ij+1} - 1, \dots; t) + \alpha_{ij}^W(o_{ij} + 1) \left(1 - \frac{o_{ij-1} - 1}{B_{ij-1}}\right) P(\dots, o_{ij-1} - 1, \dots, o_{ij} + 1, \dots; t) \\ &+ \alpha_{ij}^S(o_{ij} + 1) \left(1 - \frac{o_{i+1j} - 1}{B_{i+1j}}\right) P(\dots, o_{ij} + 1, \dots, o_{i+1j} - 1, \dots; t)\} - \sum_{i,j=1}^N \{\alpha_{ij}^N \left(1 - \frac{o_{i-1j}}{B_{i-1j}}\right) + \\ &+ \lambda_{ij} + \mu_{ij} + \alpha_{ij}^E \left(1 - \frac{o_{ij+1}}{B_{ij+1}}\right) + \alpha_{ij}^W \left(1 - \frac{o_{ij-1}}{B_{ij-1}}\right) + \alpha_{ij}^S \left(1 - \frac{o_{i+1j}}{B_{i+1j}}\right)\} o_{ij} P(\dots, o_{ij}, \dots; t) \end{aligned} \quad (5.6)$$

with the initial condition $P(o_{11} = 0, \dots, o_{ij_S} = 1, \dots, o_{NN} = 0; t = 0) = 1$ which shows that the stochastic packet dissemination was started at location $(i, j)_S$ on the mesh.

The above master equation describes the time evolution of the *probability density function* (PDF) of a packet type which follows a stochastic routing process in a finite mesh network. More precisely, each of the terms on the right hand side of Eq. 5.6 accounts for one of the above mentioned transition probabilities (*i.e.*, possible individual nodes actions) applied to a certain node (i, j) with $i, j = 1 \div N$ in the time interval $(t, t + \delta t)$. All other possible events have probability $O(\delta t)$, as δt tends to zero, and so they are negligible. Intuitively, Eq. 5.6 gives the dynamic description of the $P(o_{11}, o_{12}, \dots, o_{NN}; t)$ distribution, at microscopic-level, over *all* possible nodes configurations. Of note, as a particular case, Eq. 5.6 can describe the evolution of M independent random walks on a finite graph by neglecting the branching (λ_{ij}) and annihilation (μ_{ij}) parameters. Besides the problem at hand, this is particularly interesting for characterizing biological processes where the duplication and annihilation rates are much smaller compared to the time horizon of the entire process (*e.g.*, drug delivery via bacteria [143]).

5.4 Performance Analysis: Hitting Probabilities and Hitting Times

The mean hitting time $\langle T_{(i,j)_S \rightarrow (i,j)_D} \rangle$ can be expressed as follows:

$$\langle T_{(i,j)_S \rightarrow (i,j)_D} \rangle = \int_0^\infty t \left[\frac{\partial P(o_{(i,j)_D} = 0, t | o_{(i,j)_S} = 1, t = 0)}{\partial t} \right] dt \approx \int_0^\infty t P(o_{(i,j)_D} = 0, t) dt \quad (5.7)$$

where we use integration by parts and the fact that $P(o_{(i,j)_D} = 1, t; o_{(i,j)_S} = 1, 0)$ is much smaller than the probability $P(o_{(i,j)_D} = 0, t; o_{(i,j)_S} = 1, 0)$. Eq. 5.7 quantifies the average time needed by the first packet to reach the destination $(i, j)_D$ from a given source node $(i, j)_S$.

Due to node failures, a fault-tolerant computation strategy should also consider that a given set of distributed tasks may be implemented on multiple NoC tiles by sending the data and program instructions via a stochastic communication protocol. Consequently, one needs to estimate, in advance, the time it takes to communicate the data and the computational requests from one source to a set of destination nodes. Indeed, similarly to Eq. 5.7, the mean hitting time ($\langle T_{set} \rangle$) between a source $(i, j)_S$ and a set of destinations $\Delta = \{(i, j)_{D_k} | k \leq \text{card}(\Delta)\}$ can be estimated using the following formula:

$$\begin{aligned} \langle T_{set} \rangle = \max & \left(\int_0^\infty t P(o_{(i,j)_{D_1}} = 0, t) dt, \int_0^\infty t P(o_{(i,j)_{D_2}} = 0, t) dt, \dots, \right. \\ & \left. \int_0^\infty t P(o_{(i,j)_{D_{\text{card}(\Delta)-1}}} = 0, t) dt, \int_0^\infty t P(o_{(i,j)_{D_{\text{card}(\Delta)}}} = 0, t) dt \right) \end{aligned} \quad (5.8)$$

We evaluate the proposed model by considering a stochastic communication scenario between node $(1, 1)_S$ and node $(1, 20)_D$ on a 30×30 mesh NoC. Figure 5.4.a shows the numerical estimate of the probability of node $(1, 20)_D$ to receive exactly 0, 1, and 3 packets under stochastic communication. For this experiment, we use a 0.15 packet injection rate, and a 0.8 link successful transmission probability for all directions. We compare these results with those shown in Figure 5.4.b for the probability of not receiving any packet and the probability of receiving exactly 1 and 3 packets for the node located at $(1, 20)$ on a 30×30 mesh, from source $(1, 1)_S$ obtained via simulation. Both graphs show a similar exponential behavior of the hitting probabilities. Knowing the probability of not receiving any packet allows us to compute also the time-dependent probability of receiving at least 1 packet. This probability can be further used to determine upper-bounds on the mean hitting time. For instance, as shown in Figure 5.4.a, the destination at $(1, 20)_D$ receives at least 1 packet in approximately 23 communication rounds.

Along the lines of dynamic processes evolving on networked architectures, this hitting time analysis opens up two research directions: firstly, it enables us to estimate the performance of NoC architectures employing various routing protocols (*e.g.*, XY, adaptive, stochastic) in the presence of hardware failures; secondly, it can serve as a mathematical tool for characterizing the evolution

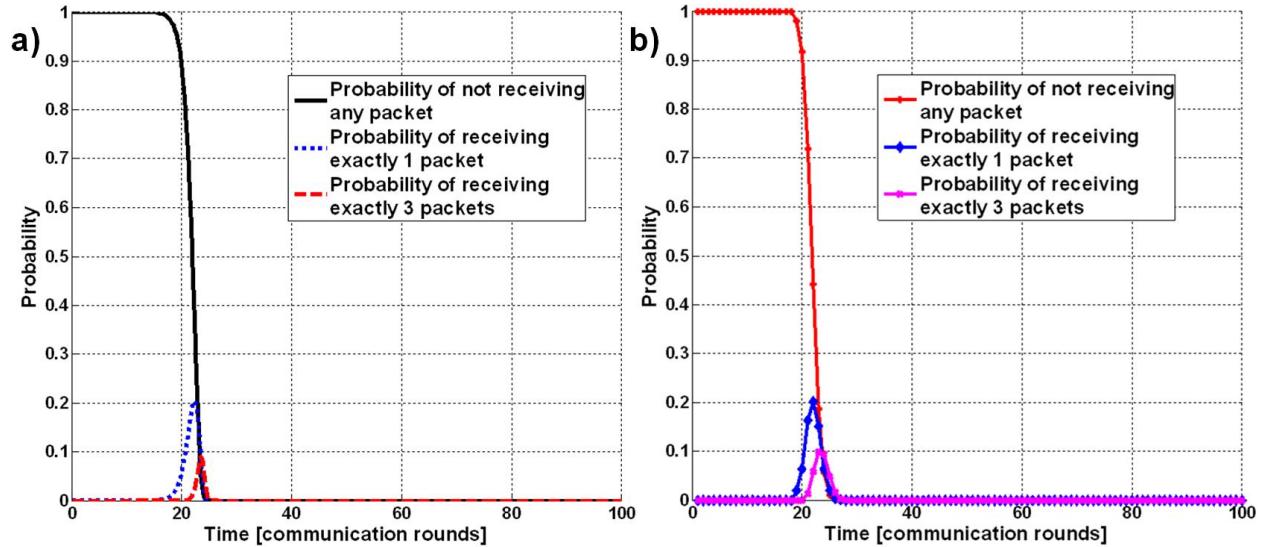


Figure 5.4: a) Time-dependent probabilities at destination (1, 20) to receive 0, 1, and 3 packets from node (1, 1) on a 30×30 mesh obtained via numerical analysis. The probability of receiving at least 1 copy is almost 1 after 23 communication rounds. b) Time-dependent probabilities for nodes (1, 20) to receive 0, 1, and 3 packets from node (1, 1) on a 30×30 mesh obtained via simulation.

of a network of swarming (nano)robots that can achieve either medical or defense tasks.

Chapter 6

Dynamic Optimization Techniques for Networked-based Architectures

6.1 The Quest for Limiting Watts Expenditure at Nanoscale

Integrating a large number of cores that need to operate at high frequencies in order to accommodate complex applications (*e.g.*, recognition, mining, and synthesis (RMS) [164], or data-centric computation [131]) leads to very heterogeneous workloads and higher power consumption and temperature fluctuations within die [22]. In this context, in order to sustain the increasing computational demands, it is essential to enhance the multicore platforms with smart power management policies, which can enable per core/tile control of power consumption while satisfying various performance levels [97].

Smart power management policies are critically needed not only because of the bursty nature of computational workloads, but also because systematic and random variations in process, voltage and temperature can significantly degrade chip reliability. Such problems cannot be predicted nor corrected at design and manufacturing stages, so it is essential to provide efficient and *robust online control methodologies* that can minimize the impact of workload dynamic variations on power consumption and peak temperature profiles, while satisfying the platform performance constraints.

To overcome these challenges, we formulate the problem of optimal power management for multi-domain platforms, where communication happens via a globally asynchronous locally synchronous (GALS) NoC architecture. As shown in Figure 6.2, the GALS NoC approach implies that the NoC is partitioned into synchronous blocks which communicated with each other asynchronously. In our framework, we assume that both PEs and routers can run at different supply and threshold voltages. We denote a block running at a certain supply and threshold voltage as a voltage-frequency island (VFI). The goal of the online controller is to determine the *optimal* operating frequencies for both PEs and NoC routers that belong to separate voltage and frequency islands (VFIs) such that the performance constraints (typically translated into queue utilization values) are met despite the highly complex characteristics (*e.g.*, non-stationarity, self-similarity) exhibited

by real computational workloads. As shown in Figure 6.1, the contribution of our work is threefold:

- First, we propose a *fractal-based state description* of the dynamics of queues interfacing neighboring VFIs (top part of Figure 6.1). For completeness, we also describe a strategy for estimating the parameters of the fractal model (middle part of Figure 1).
- Second, we formulate the power management as a *constrained finite horizon fractional optimal control problem*, which seeks to bring the utilization of the queues in the NoC platforms at predefined reference values while minimizing the individual operating frequency of both PEs and routers (see Figure 6.1). Note that the controller we synthesize accounts for the high variability observed in computational workloads, as well as ensures that the operating frequencies of both processors and routers remain within a predefined interval.
- Third, using advanced concepts like *Lagrange optimization* and *calculus of variations*, we derive the *optimality conditions* that need to be satisfied by all operating frequencies across the VFIs in order to reach the desired performance level for the entire multicore platform.

Generally speaking, feedback-based control approaches compute a set of control actions which bring the system into a desired state with no constraints on the magnitude of the control signal. In many situations, however, such control signals can take exceedingly high values which make them (physically) unfeasible. In addition, the feedback control strategies have the drawback that only a limited number of design parameters can be found from the closed-loop pole locations. An alternative approach is to consider the problem of *finite horizon optimal control*¹ with a predefined reference which finds the *best sequence of control actions over a fixed time interval* (horizon); this set of control actions can bring the system (characterized by an integer order differential equation) to the desired reference at the end of the control interval.

In contrast to feedback control approaches and classical optimal control, the (finite horizon) fractional optimal control approach investigated here allows to directly optimize a certain performance function subject to fractal (*i.e.*, fractional derivatives) state equations (*i.e.*, for queue utilization) and bounded control signals (*i.e.*, operating frequencies). In other words, the proposed fractal controller is able to provide the optimal control signals (*i.e.*, operating frequencies), if they exist, for a given performance level. In contrast, the classical feedback control approaches (based on integer order derivatives) can determine operating frequencies that do not necessarily offer the minimum power consumption.

¹Note that, in control theory, the finite horizon optimal control problem, which minimizes a specific performance index function while selecting control signals from a constrained set, is also referred as robust receding horizon control or as robust model predictive control.

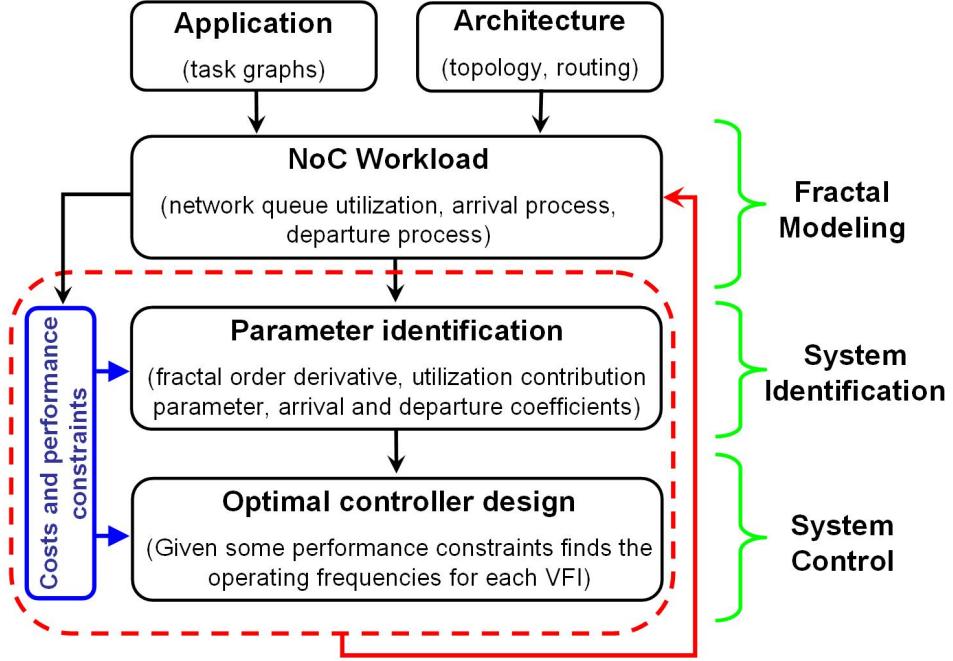


Figure 6.1: Overview of our methodology for controlling fractal workloads in multi-core systems. We first input the information about the application (*e.g.*, task graphs, packet injection rates) and architecture (*e.g.*, topology, routing) and the set of costs and performance constraints that have to be satisfied by the NoC platform. Building on NoC workload measurements (*i.e.*, queue utilization, arrival and departure processes), we build a fractal model of the queue dynamics by estimating the fractional order of the time derivative, the utilization contribution parameter and the arrival and departure coefficients. Next, we use this identified model to design an optimal controller, which determines the optimal operating frequencies for the VFIs such that the performance constraints are satisfied.

6.2 Power Management for NoC Architectures under Highly Variable Workloads

We formulate the power management as an optimal control problem which takes into account the fractal characteristics of the NoC workload. Towards this end, we consider a VFI-based NoC architecture consisting of N_{PE} PEs, N_r routers, and N_j^q queues interfacing the router in the j -th VFI with other routers in the neighboring VFIs (see Figure 6.2).

The goal of our nonlinear control problem is to find, for a given starting time (t_i) and a final time (t_f), the optimal assignment of operating frequencies for the PEs, routers, and queues, which minimizes a multiobjective function consisting of the quadratic costs of queues utilization with respect to a predefined reference, as well as, the operating frequency of each VFI (this would

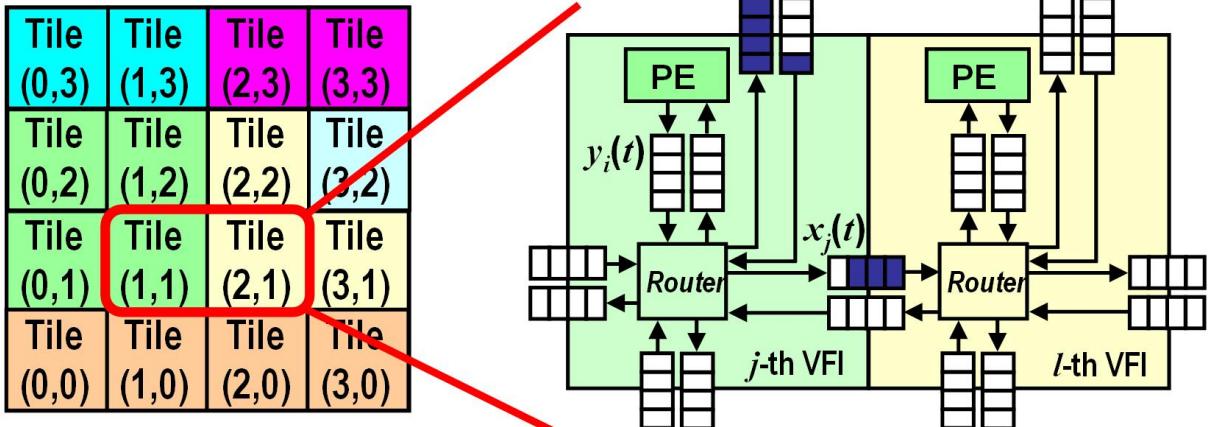


Figure 6.2: Representation of j -th and l -th neighboring VFIs where each PE is set to run, if necessary, at its own frequency. The $x_j(t)$ variable represents the utilization of the interface queue between the j -th and l -th VFIs. The $y_i(t)$ represent the utilization values of the interface queue between the i -th PE and j -th VFI. Note that various colors of the tiles in the above NoC imply that each island can run at a certain frequency.

implicitly minimize also the power consumption):

$$\min \int_{t_i}^{t_f} \left\{ \sum_{i=1}^{N_{PE}} \frac{w_i (y_i(t) - y_i^{ref}(t))^2 + z_i f_i^2(t)}{2} + \sum_{j=1}^{N_r} \sum_{k=1}^{N_j^q} \frac{r_j f_j^2(t) + q_k (x_k(t) - x_k^{ref})^2}{2} \right\} dt \quad (6.1)$$

subject to the constraints given in Eq. 6.2 through Eq. 6.6:

$$\frac{d^{\alpha_i} y_i(t)}{dt^{\alpha_i}} = a_i(t) y_i(t) + b_i(t) f_i(t) - c_i(t) f_j(t), \quad i = 1 \div N_{PE} \quad (6.2)$$

$$0 \leq y_i^{min} \leq y_i(t) \leq y_i^{max} \leq 1, \quad i = 1 \div N_{PE} \quad (6.3)$$

where $y_i(t)$ and $y_i^{ref}(t)$ for $i = 1 \div N_{PE}$ represent the actual utilization and the utilization reference of the queue between the i -th PE and its corresponding router, $x_k(t)$ and $x_k^{ref}(t)$ for $k = 1 \div N_j^q, j = 1 \div N_r$ are the actual utilization and the reference utilization of the k -th queue between the routers in the j -th and l -th VFIs. In Eq. 6.1, w_i , z_i , r_j and q_k are positive weighting coefficients. In Eq. 6.2, α_i is the fractional order which depends on the fractal dimension characterizing the queue utilization process $y_i(t)$, a_i represents the weight coefficient of the utilization $y_i(t)$, b_i and c_i reflect the contributions of the writing frequency (f_i) and the reading frequency (f_j), y_i^{min} and y_i^{max} are the admissible lower and upper bounds on the queue utilization $y_i(t)$. Of note, the optimal controller allows the designer to set individual weighting coefficients (*i.e.*, w_i , z_i , r_j and q_k) in Eq. 6.1 for each of the NoC components such that the major power consumption elements can have a higher impact on the overall cost function.

The next set of constraints are meant to characterize the utilization of queues between neigh-

boring VFIs:

$$\frac{d^{\alpha_k} x_k(t)}{dt^{\alpha_k}} = a_k(t)x_k(t) + b_k(t)f_k(t) - c_k(t)f_l(t), k = 1 \div N_j^q, j = 1 \div N_r \quad (6.4)$$

$$0 \leq x_k^{min} \leq x_k(t) \leq x_k^{max} \leq 1, k = 1 \div N_j^q, j = 1 \div N_r \quad (6.5)$$

where α_k is the fractional order characterizing the queue utilization process $x_k(t)$, $a_k(t)$ represents the contribution of utilization $x_k(t)$ to the entire queues utilization dynamics, $b_k(t)$ and $c_k(t)$ represent coefficients of the writing frequency (f_k) and the reading frequency (f_l), respectively, x_k^{min} and x_k^{max} are the lower and upper bounds on the utilization $x_k(t)$.

Although in current problem formulation we assumed that workload parameters (*i.e.*, a_i , b_i and c_i related to the dynamical equation of the utilization $y_i(t)$; $a_k(t)$, $b_k(t)$ and $c_k(t)$ related to the dynamical equation of the utilization $x_k(t)$) are taking deterministic values, the proposed formalism can also be extended to account for uncertainty coming from either the estimation procedure of these parameters or from system non-idealities (*e.g.*, process variations). If the stochastic nature of the model parameters can be modeled through Gaussian type of distributions, then the problem formulation needs to be augmented with lower and upper bounds to account for the variation exhibited by each parameter. This represents a straight forward extenssion of present formalism by considering that each fractal state-space equation (see Eq. 6.2 or Eq. 6.4) is replaced by two dynamical constraints corresponding to the lower and upper bounds on the parameter. If the parameter uncertainty of the dynamical equations (*i.e.*, Eq. 6.2 or Eq. 6.4) exhibits non-Gaussian characteristics, then the controller needs to be synthesized via solving a fractional Hamilton-Jacoby type of partial differential equation described in [30].

Note that the cost function in Eq. 6.1 seeks to maintain *all* NoC queues at specific utilization references (see the squared differences between $y_i(t)$ and $y_i^{ref}(t)$ or $x_k(t)$ and $x_k^{ref}(t)$), while the control inputs - the operating frequencies - are prevented from taking exceedingly large values, which would correspond to high power consumption. In other words, the role of the optimal controller is to select the *minimum operating frequencies* for which the performance constraints are satisfied.

Moreover, in order to prevent the nonlinear controller from selecting an unacceptable range of operating frequencies, we impose the following bounding constraints:

$$f_i^{min}(t) \leq f_i(t) \leq f_i^{max}(t), i = 1 \div N_{PE}, f_j^{min} \leq f_j \leq f_j^{max}, j = 1 \div N_r \quad (6.6)$$

where f_i^{min} and f_i^{max} are the lower and upper bounds on the frequency at which each PE can run, f_j^{min} and f_j^{max} are the lower and upper bounds on the frequency at which each router can run. Note that we introduce two indices i - for the PEs and j - for the routers, and implicitly two variables (*i.e.*, f_i and f_j) such that the operating frequencies of the PEs are decoupled from the router frequencies. Hence, we avoid setting the PE to a small frequency which may affect the computational performance requirements, or setting the router to a too high frequency when it is

not necessary. In addition, by setting bounds on operating frequency can also help at mitigating circuit aging and nano-scale challenges (*e.g.*, parameter variations).

Although in the above problem formulation we consider that the queues attached to a router and their corresponding router have the same operating frequency, the mathematical formulation can be extended to account for distinct voltage-frequency domains for routers and buffers. Since considering a single VFI for each router would introduce further complexity (due to a larger number of mixed-clock queues), we limit ourselves at considering that the N_r consist of just a few VFIs and include more constraints to reflect the fact that neighboring routers are assumed to operate at the same frequency. Nevertheless, there is a tradeoff between the computational time required to solve the power management problem and determine the optimal operating frequencies and the number of VFIs. In the following section, we show how the optimization problem can be solved via calculus of variations and optimal control theory concepts, derive the optimality conditions and summarize the computational complexity of the proposed power management algorithm.

6.3 Algorithmic Perspective and Experimental Results

To solve the power optimization problem, we use concepts from the optimization theory and construct first the Lagrangian functional, $L(y_i, f_i, \lambda_i, x_k, f_j, \gamma_{k,j})$, as follows:

$$L(y_i, f_i, \lambda_i, x_k, f_j, \gamma_{k,j}) = \int_{t_i}^{t_f} \left\{ \sum_{i=1}^{N_{PE}} \left[\frac{w_i}{2} \left(y_i(t) - y_i^{ref}(t) \right)^2 + \frac{z_i f_i^2(t)}{2} + \lambda_i \left(\frac{d^{\alpha_i} y_i(t)}{dt^{\alpha_i}} - \right. \right. \right. \\ \left. \left. \left. - a_i(t) y_i(t) - b_i(t) f_i(t) + c_i(t) f_j(t) \right) \right] + \sum_{j=1}^{N_r} \left[\frac{r_j f_j^2(t)}{2} + \sum_{k=1}^{N_j^q} \frac{q_k}{2} \left(x_k(t) - x_k^{ref} \right)^2 + \right. \\ \left. \left. \left. \gamma_{k,j} \left(\frac{d^{\alpha_k} x_k(t)}{dt^{\alpha_k}} - a_k(t) x_k(t) - b_k(t) f_k(t) + c_k(t) f_l(t) \right) \right] \right\} dt \quad (6.7)$$

where $y_i(t)$ and $x_k(t)$ denote the queue utilization variables, f_i is the frequency associated with the i -th PE, f_j is the frequency associated with the j -th router, λ_i is the Lagrange multiplier associated with the constraint imposed for the queue between the PE and the router, and $\gamma_{k,j}$ are the Lagrange multipliers associated with the constraints imposed on the queue between neighboring routers in different VFIs.

For completeness, we also have to add some boundary constraints on the utilization of mixed clock queues:

$$y_i(t=0) = y_i^0 \quad y_i(t=t_f) = y_i^{ref}, \quad i = 1 \div N_{PE} \quad (6.8)$$

$$x_k(t=0) = x_k^0 \quad x_k(t=t_f) = x_k^{ref}, \quad k = 1 \div N_j^q, \quad j = 1 \div N_r \quad (6.9)$$

These conditions are required in order to satisfy a certain performance level from the compu-

tation standpoint.

By expanding the Lagrange function in Eq. 6.7 via the Taylor formula and considering that it attains its minimum in the vicinity of $\tau = 0$, *i.e.*, $\partial L/\partial\tau = 0$, we obtain the following relations:

$$\frac{\partial L}{\partial y_i} + {}_t D_{t_f}^{\alpha_i} \frac{\partial L}{\partial {}_{t_i} D_t^{\alpha_i} y_i} = 0, \quad \frac{\partial L}{\partial f_i} = 0, \quad i = 1 \div N_{PE} \quad (6.10)$$

$$\frac{\partial L}{\partial x_k} + {}_t D_{t_f}^{\alpha_k} \frac{\partial L}{\partial {}_{t_i} D_t^{\alpha_k} x_k} = 0, \quad \frac{\partial L}{\partial f_j} = 0, \quad k = 1 \div N_j^q, \quad j = 1 \div N_r \quad (6.11)$$

where ${}_t D_t^\alpha$ and ${}_t D_{t_f}^\alpha$ represent the fractional derivatives of order α operating backward and forward in time [80], respectively.

In order to solve the equations in Eq. 6.10, we discretize the interval $[t_i, t_f]$ into N intervals of size $(t_f - t_i)/N$ and use the formula in Eq. 1.4 to construct a linear system which can be solved using either Gaussian elimination or LU decomposition. In short, the algorithm of the optimal controller is as follows:

Algorithm for optimal control synthesis:

1. Read from the system identification module, the coefficients α_i , a_i , b_i , c_i for $i = 1 \div N_{PE}$ characterizing the dynamics of queues between PEs and routers and α_k , a_k , b_k , c_k for $k = 1 \div N_j^q, j = 1 \div N_r$ describing the dynamics of queues between neighboring routers in different VFIs;
2. For a fixed number of discrete steps N , compute the coefficients obtained after discretization of the fractional derivatives in Eq. 6.2, Eq. 6.4 and Eq. 6.10 using the formula in Eq. 1.4 and construct a linear system, where the unknown variables are represented by the operating frequencies (*i.e.*, f_i , f_j) and Lagrange multipliers (*i.e.*, λ_i and $\gamma_{k,j}$);
3. Solve the linear system in Eq. 6.10 and find the operating frequencies for each VFI in the NoC architecture.

In terms of practical implementation, the worse-case complexity² of the algorithm for the case, when minimum and upper bounds on both queues utilization and operating frequency are considered, is bounded by $O(N \times M^2)$ where N is the number of discretization steps and $M = 2N_{PE} + 2 \sum_{j=1}^{N_r} N_j^q$ is the sum between the number of state variables (*i.e.*, queues utilization) and the number of control signals (*i.e.*, operating frequencies). However, if the minimum and maximum bound constraints on the queues utilization and operating frequency are ignored, then the controller algorithm reduces to a linear system in Eq. 6.10 with a linear complexity $O(N + M)$ when solved by $(N + M)^2$ adders. This low complexity makes it perfectly suitable for online implementation.

To evaluate the fractal optimal control, we consider a combination of trace driven and cycle accurate simulation of a VFI-based NoC architecture. From an application perspective, we consider four

²Note that here we refer to worse-case time complexity [79][141] which represents the number of discrete steps required to solve the formulated optimization problem on a universal Turing machine [152]. The worse-case space complexity (*i.e.*, the number of distinct storage locations accessed by various instructions required by the optimization algorithm), can be reduced by using hash tables and employing the multiparametric linear programming methods of partitioning the state-input space of polyhedral regions.

Workload	Description
apache	Web server application (apache HTTP server v2.0): 16K connections, worker threading model
db_tpcc	Online transaction processing (IBM DB2 v8 ESE): 100 warehouses, 64 clients, 450 MB buffer pool
oracle	Online transaction processing (POracle 10g Enterprise Database Server): 100 warehouses, 16 clients, 1.4 GB SGA
ocean	Scientific (ocean simulation): 1026×1026 grid, 9600 relaxations, 10^{-7} error tolerance
sparse	Scientific (blocked sparse Cholesky factorization): 4096x4096 matrix

Table 6.1: Commercial and scientific workloads used to test the constrained finite horizon fractal optimal control approach to power and peak temperature management.

16-node multi-threaded commercial workloads (i.e., *Apache HTTP server v2.0* from SPECweb99 benchmark [146], online transaction processing application consisting of *TPC-C v3.0 workload* on both *IBM DB2 v8 ESE* and *Oracle 10gEnterprise Database Server*, *blocked sparse Cholesky factorization* and *ocean simulation*) obtained by running them on a FLEXUS based shared-memory 16-processor environment consisting of cycle accurate models of out-of-order processors and cache hierarchy [160][163] (see also Table 6.1 for more details).

From an architectural perspective, we consider a 4×4 mesh NoC employing XY wormhole routing scheme with mixed clock queues of 10 flit size and packets consisting of 15 flits. In this setup, we consider that the execution of a set of applications is divided into several intervals of 20ms length. Nevertheless, the proposed control algorithm can also work with larger time intervals while considering that the model parameters are estimated for the new time scale. For each time interval, the system identification module estimates the fractional exponent α_k in two stages: First it computes the wavelet coefficients for $\log_2(m)$ resolution scales via the discrete wavelet transform of the queue utilizations (of size m). Second, it performs a linear regression between the resolution scales and the variance of the wavelets coefficients. This parameter identification strategy not only reduces the computational complexity from $O(N^3)$ order to a linear $O(N)$, but also allows online estimation procedure with minimum memory overhead (i.e., it does not require to store all queue utilizations and can be performed iteratively whenever there is a change in queue occupancy). Next, the identification module estimates from the arrival (A_k), departure (D_k) and queue utilization (X_k) processes the parameters a_k , b_k , and c_k by solving the following linear system:

$$\begin{bmatrix} Re[X_k(j\omega_1)] & Re[A_k(j\omega_1)] & -Re[D_k(j\omega_1)] \\ 0 & Im[A_k(j\omega_1)] & -Im[D_k(j\omega_1)] \\ \dots & \dots & \dots \\ Re[X_k(j\omega_M)] & Re[A_k(j\omega_M)] & -Re[D_k(j\omega_M)] \\ 0 & Im[A_k(j\omega_M)] & -Im[D_k(j\omega_M)] \end{bmatrix} \begin{bmatrix} a_k \\ b_k \\ c_k \end{bmatrix} = \begin{bmatrix} \Lambda(\omega_1) \\ \Upsilon(\omega_1) \\ \dots \\ \Lambda(\omega_M) \\ \Upsilon(\omega_M) \end{bmatrix} \quad (6.12)$$

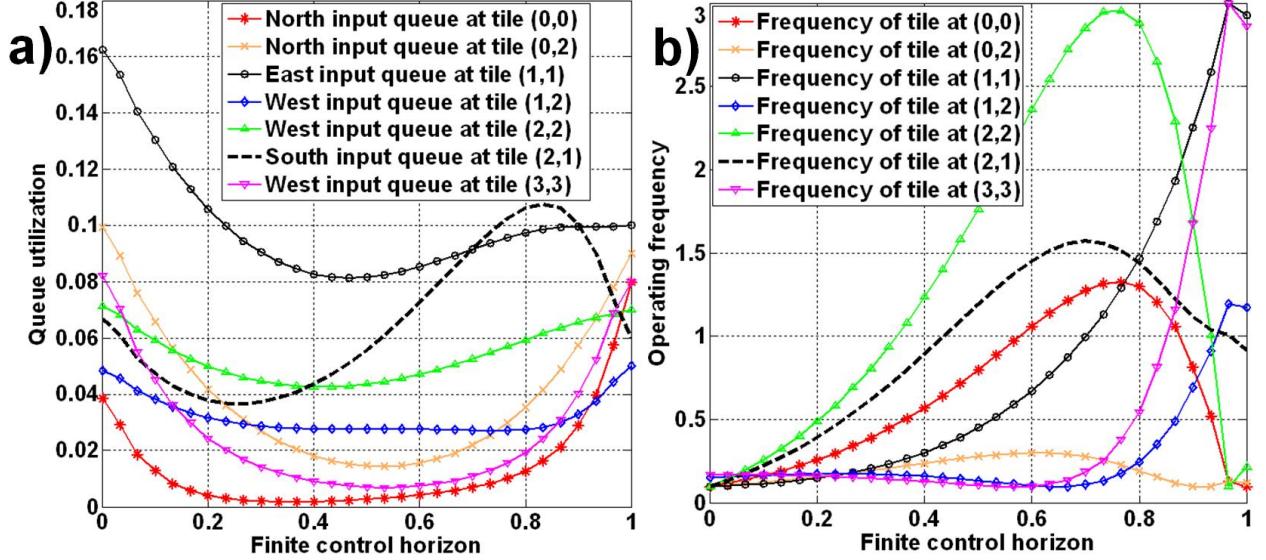


Figure 6.3: a) Utilization of the queues at tiles (0,0), (0,2), (1,1), (1,2), (2,2), (2,1), and (3,3) for a 4×4 mesh NoC running Apache HTTP webserver application. b) The variation of the operating frequencies for the routers at (0,0), (0,2), (1,1), (1,2), (2,2), (2,1), and (3,3) necessary to reach the reference values imposed on the utilization of all queues.

where the terms $\Lambda(\omega_1) = \omega_1^{\alpha_k} \cos(\alpha_k \pi / 2) \text{Re}[X_k(j\omega_1)]$ and $\Upsilon(\omega_1) = \omega_1^{\alpha_k} \sin(\alpha_k \pi / 2) \text{Im}[X_k(j\omega_1)]$ come from the frequency representation of Eq. 6.4. After the identification step is completed (at run-time) in parallel with the application computations, the fractal optimal controller reads these parameters and solves the linear system defined by Eq. 6.10 to determine the optimal operating frequencies for the PEs and routers (for the next interval of 20ms) that ensure a predefined performance level specified in terms of queue utilization references.

Following the above procedure, we apply the optimal controller for $N = 30$ discrete steps and a 4×4 mesh NoC running a *Apache HTTP webserver* application to determine the operating frequencies such that the utilization of the queues is below 0.1. The reason for bounding the maximum utilization of the queues at a reference value of 0.1 is two fold: First, a smaller utilization of the queues implies smaller packet waiting times in buffers and so smaller source-to-destination latencies. Second, a small queue utilization also minimizes the chances of thermal hotspot buildup and so improves chip reliability. Note also that the number of discrete steps (N) chosen for discretizing the Eq. 6.2, 6.4 and Eq. 6.10 influences the precision of the operating frequencies. Consequently, when less precision is needed in terms of operating frequency, we can use a smaller number of discrete steps (*e.g.*, 5 to 10). For demonstration purposes, we consider the case of $N = 30$ discrete steps and solve the linear system describing the dynamics of 80 queues in less than 200ns. This makes the proposed approach suitable for online power management of future multicore platforms.

Figure 6.3.a shows the utilization of a few queues at tiles (0,0), (0,2), (1,1), (1,2), (2,2), (2,1), and (3,3). We can observe that the optimal controller is able to bring the utilization these

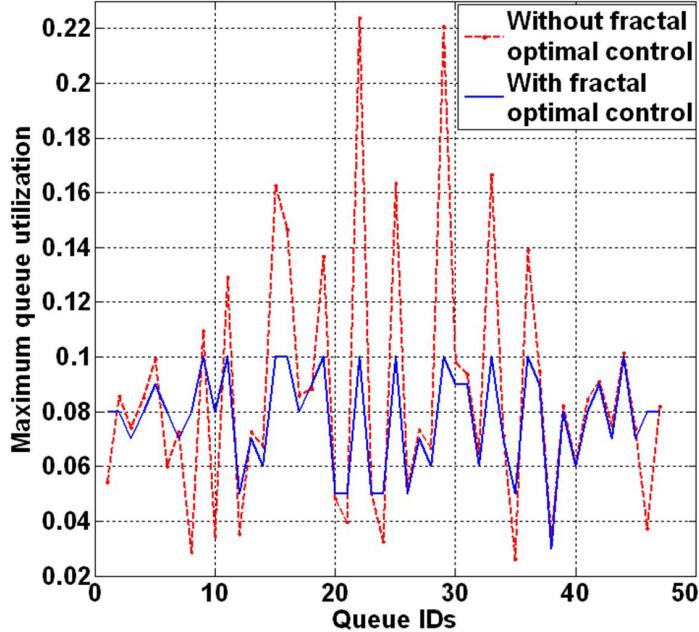


Figure 6.4: Comparison between the utilization of all queues in the uncontrolled case and the queue utilization for the case of the fractal optimal controller.

tiles below the reference value of 0.1. Figure 6.3.b shows the operating frequencies of the tiles at $(0,0), (0,2), (1,1), (1,2), (2,2), (2,1)$, and $(3,3)$ needed to attain the imposed reference values. Moreover, Figure 6.4 shows the utilization of the queues without and with fractal optimal control. The optimal controller is able to keep the utilization of all queues below 0.1 by adjusting the operating frequencies of all PEs and routers.

By comparing the power consumption of the new VFI system with an NoC architecture with all PEs and routers running at $3GHz$ we obtain approximately 70% power savings. Of note, if we restrict our optimal controller to use only integer order derivatives like in the case of linear quadratic regulator (LQR), and thus model the queue utilization processes in the NoC via integer order differential equations, the power savings are only about 30%. This shows that in some cases the classical LQR approach can get trapped in some local minima. However, the fractal optimal controller allows to find the optimal solution; this enables the highest amount of power savings.

For completeness, we also apply the optimal controller to a 4×4 mesh NoC running a matrix Cholesky factorization type of application with the goal of keeping the utilization of all queues under 0.4 utilization. The reason for constraining the utilization of all queues to be below 0.4 is to minimize the chances of thermal hotspots while allowing a prescribed performance level in terms of network throughput. Note that, by imposing at most 0.4 queue utilization for this experimental setup, the impact on average packet latency is less than 4%.

As we can see from Figure 6.5.a the utilization of the queues at tiles $(1,1), (1,3), (2,0), (2,2)$, and $(3,3)$, respectively, are brought under 0.4 at predefined reference values. Figure 6.5.b shows the operating frequencies at tiles $(0,0), (1,1), (1,3), (2,2), (2,3)$, and $(3,3)$ needed for reaching the

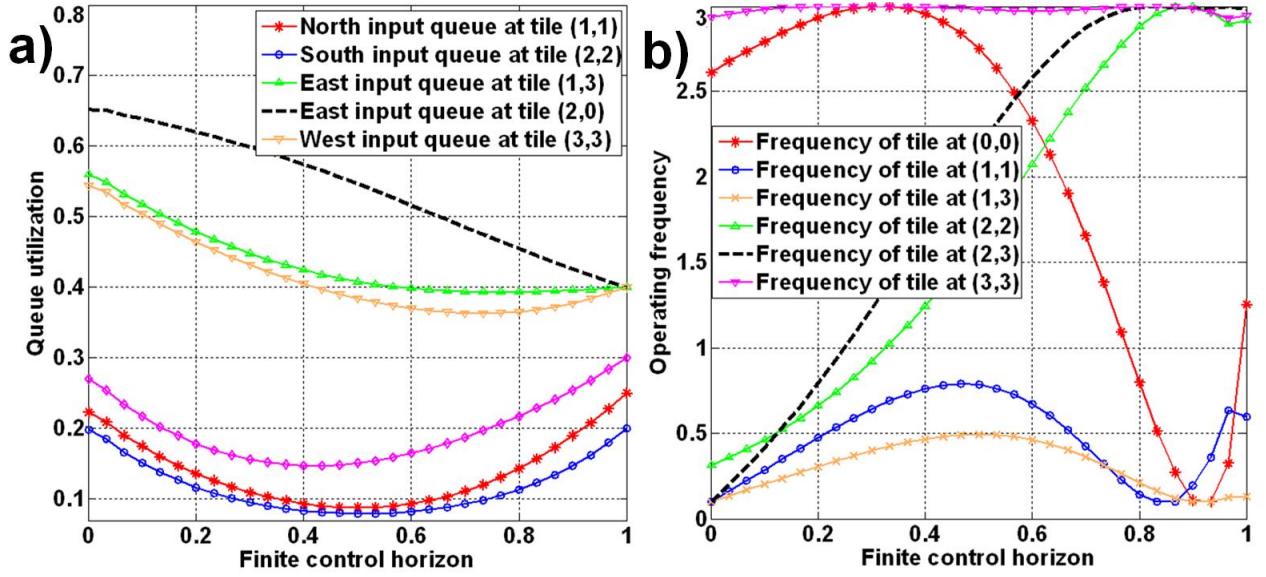


Figure 6.5: a) Utilization of the queues at tiles (1,1), (1,3), (2,0), (2,2), and (3,3), respectively, for a 4×4 mesh NoC running the Cholesky matrix factorization. b) The operating frequencies for tiles at (0,0), (1,1), (1,3), (2,2), (2,3) and (1,3) necessary for all queues in the network to meet the imposed constraints.

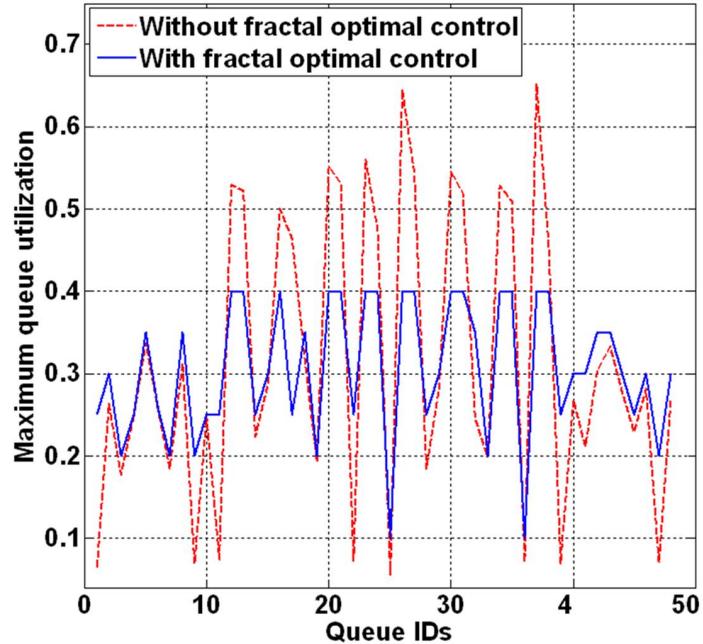


Figure 6.6: Comparison in terms of queue utilization between uncontrolled and the control case.

predefined utilization reference values.

To better emphasize the role of control, Figure 6.6 shows the utilization of all queues in the network for two cases: when no control is used (red dotted line) and when the proposed optimal controller is applied (blue solid line). We can observe that although in some cases the queue utilization increases, the peaks exceeding 0.4 are minimized by carefully assigning operating frequencies for all PEs and routers. In addition by solving the linear system in Eq. 6.10 for 30 discrete steps and comparing the power values with those of a homogeneous NoC with PEs and routers running at 3GHz, we observe that the proposed approach leads to 40% power savings for Cholesky matrix factorization. By applying the proposed algorithm and comparing with NoC architectures running at 3GHz lead to 50% and 20% power savings for ocean simulation and online transaction processing application, respectively.

So far, we investigated how we can use concepts from optimal control and fractional calculus to minimize dynamic power consumption. Next, we will consider both dynamic and static power consumption and peak temperature as explicit terms in the cost function of the dynamic optimization problem.

6.4 Power and Peak Temperature Optimization of NoCs

To account for thermal effects, in this section, we formulate the total power and thermal optimization as a constrained finite horizon fractal optimal control problem which takes into account

- i) the fractal characteristics of the NoC workload and
- ii) the nonlinearity dependency between the chip temperature and operating frequency.

We consider an $M \times N$ mesh VFI-based NoC architecture consisting of PEs, routers and interface queues (see, in Figure 6.7, the North, East, West, South and Local queues interfacing the router at (i, j) location on the mesh with the routers at $(i + 1, j)$, $(i, j + 1)$, $(i, j - 1)$, $(i - 1, j)$ and the local PE, respectively).

The goal of our nonlinear controller is to find, for a given starting time (t_i) and a final time (t_f), the right (best) assignment of supply voltage and operating frequencies for the PEs, routers and queues which minimizes the total power consumption, peak temperature profile and the quadratic difference between the actual queues utilization and a predefined reference:

$$\begin{aligned} \min & \int_{t_i}^{t_f} \sum_{i,j=1}^{M,N} \left[\frac{q_{ij}}{2} \left(y_{ij}(t) - y_{ij}^{ref}(t) \right)^2 + \frac{r_{ij,PE} f_{ij,PE}^2(t)}{2} + P_{ij,PE}^{dyn}(t) + P_{ij,PE}^{stat}(t) + PT_{ij,PE}(t) + \right. \\ & \quad \left. + \frac{r_{ij} f_{ij}^2(t)}{2} + P_{ij,r}^{dyn}(t) + P_{ij,r}^{stat}(t) + PT_{ij,r}(t) + \right. \\ & \quad \left. + \sum_{k=0}^4 \left(w_{ij,k,q} \left(\frac{q_{ij}^k}{2} \left(x_{ij}^{k,ref}(t) - x_{ij}^{k,ref}(t) \right)^2 + P_{ij,k,q}^{dyn}(t) + P_{ij,k,q}^{stat}(t) + PT_{ij,k,q}(t) \right) \right) \right] dt \end{aligned} \quad (6.13)$$

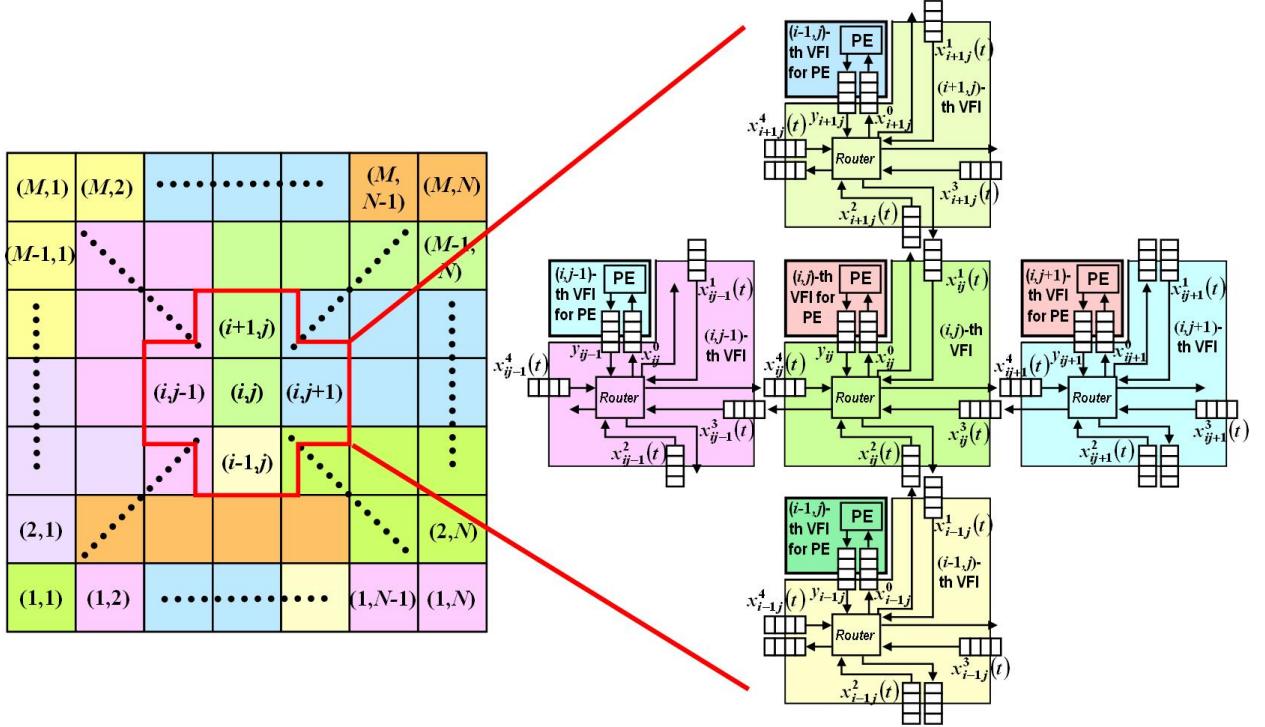


Figure 6.7: a) Representation of an $M \times N$ mesh NoC where the color of each tile shows that it belongs to a certain VFI. b) Representation of tile at (i, j) location on the mesh containing a PE which can run at a certain supply voltage and frequency, a router and five queues connecting it with the neighboring tiles.

such that the following constraints are satisfied:

$$\frac{d^{\alpha_{ij}^{PE}} y_{ij}(t)}{dt^{\alpha_{ij}^{PE}}} = a_{ij}^{PE}(t)y_{ij}(t) + b_{ij}^{PE}(t)f_{ij,PE}(t) - c_{ij}^{PE}(t)f_{ij}(t), \quad i = 1 \div M, j = 1 \div N \quad (6.14)$$

$$0 \leq y_{ij}^{min} \leq y_{ij}(t) \leq y_{ij}^{max} \leq 1, \quad i = 1 \div M, j = 1 \div N \quad (6.15)$$

where $y_{ij}(t)$ and y_{ij}^{ref} (for $i = 1 \div M$ and $j = 1 \div N$) represent the actual and reference utilization for the queue between the PE and its corresponding router at location (i, j) on the mesh, α_{ij}^{PE} is the fractional order which depends on the fractal dimension characterizing the utilization process $y_{ij}(t)$, $a_{ij}^{PE}(t)$ represents the contribution of the utilization $y_{ij}(t)$, $b_{ij}^{PE}(t)$ and $c_{ij}^{PE}(t)$ reflect the contributions of the writing frequency ($f_{ij,PE}$) and the reading frequency (f_{ij}), y_{ij}^{min} and y_{ij}^{max} are the admissible lower and upper bounds on the queue utilization $y_{ij}(t)$,

$$\frac{d^{\alpha_{ij}^0} x_{ij}^0(t)}{dt^{\alpha_{ij}^0}} = a_{ij}^0(t)x_{ij}^0(t) + b_{ij}^0(t)f_{ij}(t) - c_{ij}^0(t)f_{ij,PE}(t), \quad i = 1 \div M, j = 1 \div N \quad (6.16)$$

$$0 \leq x_{ij}^{0,min} \leq x_{ij}^0(t) \leq x_{ij}^{0,max} \leq 1, \quad i = 1 \div M, j = 1 \div N \quad (6.17)$$

$$\frac{d^{\alpha_{ij}^1} x_{ij}^1(t)}{dt^{\alpha_{ij}^1}} = a_{ij}^1(t)x_{ij}^1(t) + b_{ij}^1(t)f_{i+1j}(t) - c_{ij}^1(t)f_{ij,PE}(t), i = 1 \div M, j = 1 \div N \quad (6.18)$$

$$0 \leq x_{ij}^{1,min} \leq x_{ij}^1(t) \leq x_{ij}^{1,max} \leq 1, i = 1 \div M, j = 1 \div N \quad (6.19)$$

$$\frac{d^{\alpha_{ij}^2} x_{ij}^2(t)}{dt^{\alpha_{ij}^2}} = a_{ij}^2(t)x_{ij}^2(t) + b_{ij}^2(t)f_{i-1j}(t) - c_{ij}^2(t)f_{ij,PE}(t), i = 1 \div M, j = 1 \div N \quad (6.20)$$

$$0 \leq x_{ij}^{2,min} \leq x_{ij}^2(t) \leq x_{ij}^{2,max} \leq 1, i = 1 \div M, j = 1 \div N \quad (6.21)$$

$$\frac{d^{\alpha_{ij}^3} x_{ij}^3(t)}{dt^{\alpha_{ij}^3}} = a_{ij}^3(t)x_{ij}^3(t) + b_{ij}^3(t)f_{ij+1}(t) - c_{ij}^3(t)f_{ij,PE}(t), i = 1 \div M, j = 1 \div N \quad (6.22)$$

$$0 \leq x_{ij}^{3,min} \leq x_{ij}^3(t) \leq x_{ij}^{3,max} \leq 1, i = 1 \div M, j = 1 \div N \quad (6.23)$$

$$\frac{d^{\alpha_{ij}^4} x_{ij}^4(t)}{dt^{\alpha_{ij}^4}} = a_{ij}^4(t)x_{ij}^4(t) + b_{ij}^4(t)f_{ij-1}(t) - c_{ij}^4(t)f_{ij,PE}(t), i = 1 \div M, j = 1 \div N \quad (6.24)$$

$$0 \leq x_{ij}^{4,min} \leq x_{ij}^4(t) \leq x_{ij}^{4,max} \leq 1, i = 1 \div M, j = 1 \div N \quad (6.25)$$

where $x_{ij}^k(t)$ and $x_{ij}^{k,ref}(t)$ (for $i = 1 \div M, j = 1 \div N$ and $k = 0 \div 4$) are the actual and reference utilization for the queue between the router at the (i, j) location and the local PE for $k = 0$, the router at $(i + 1, j)$ location for $k = 1$, the router at $(i - 1, j)$ location for $k = 2$, the router at $(i, j + 1)$ location for $k = 3$, and the router at $(i, j - 1)$ location for $k = 4$, respectively. The α_{ij}^k for $k = 0 \div 4$ are the fractional order characterizing the utilization process $x_{ij}^k(t)$; $a_{ij}^k(t)$ for $k = 0 \div 4$ denote the contribution of utilization $x_{ij}^k(t)$, $b_{ij}^k(t)$ and $c_{ij}^k(t)$ for $k = 0 \div 4$ represent the contribution coefficients of the writing frequency (f_{ij} for $k = 0$, f_{i+1j} for $k = 1$, f_{i-1j} for $k = 2$, f_{ij+1} for $k = 3$, f_{ij-1} for $k = 4$) and the reading frequency (f_{ij}), x_{ij}^{min} and x_{ij}^{max} are the lower and upper bounds on the queue utilization $x_{ij}^k(t)$.

Note that the $w_{ij,k,q}$ parameter in Eq. 6.13 is used to select based on topological information only the directional queues actually connected to the router (*i.e.*, for tile at $(i, j) = (1, 1)$ location for $q = 0, 1, 3$, for tile at $(i, j) = (2, 1)$ location for $q = 0, 1, 2, 3$, and for tile in the middle of the for $q = 0, 1, 2, 3, 4$).

The dynamic power consumption terms introduced in Eq. 6.13 (for each PE, each router and each queue) are defined as follows:

$$\begin{aligned} P_{ij,PE}^{dyn} &= C_L^{PE} V_{ij,PE}^2 f_{ij,PE} S_{w_{ij,PE}} \\ P_{ij,r}^{dyn} &= C_L^r V_{ij}^2 f_{ij} S_{w_{ij,r}} \\ P_{ij,k,q}^{dyn} &= C_L^q V_{ij}^2 f_{ij} S_{w_{ij,k,q}} \end{aligned} \quad (6.26)$$

where C_L^{PE} , C_L^r and C_L^q are the load capacitances of the PE, router and queues at (i, j) location, $V_{ij,PE}$ and V_{ij} are the supply voltages of the PE and router at (i, j) location, $f_{ij,PE}$ and f_{ij} are the operating frequencies of the PE and router at (i, j) location, $S_{w_{ij,PE}}$, $S_{w_{ij,r}}$ and $S_{w_{ij,k,q}}$ are the switching activities of the PE, the router and the queues at (i, j) location.

The static power consumption terms introduced in Eq. 6.13 are defined as follows:

$$\begin{aligned} P_{ij,PE}^{stat} &= (1 + m_{ij,PE}) K_{ij,PE} (k_B T_{ij,PE})^2 V_{ij,PE} e^{-V_{ij,PE}^{th}/S_{ij,PE}} / q_e^2 \\ P_{ij,r}^{stat} &= (1 + m_{ij,r}) K_{ij,r} (k_B T_{ij,r})^2 V_{ij} e^{-V_{ij}^{th}/S_{ij,r}} / q_e^2 \\ P_{ij,k,q}^{stat} &= (1 + m_{ij,k,q}) K_{ij,k,q} (k_B T_{ij,k,q})^2 V_{ij} e^{-V_{ij}^{th}/S_{ij,k,q}} / q_e^2 \end{aligned} \quad (6.27)$$

where $m_{ij,PE}$, $m_{ij,r}$ and $m_{ij,k,q}$ are the leakage current dissipation parameter (dependent on the maximum depletion layer capacitance of the semiconductor under the oxide and the oxide capacitance) of the PE, router and queues at (i, j) location, k_B is the Boltzmann constant, $K_{ij,PE}$, $K_{ij,r}$ and $K_{ij,k,q}$ are the device transconductance [9] of the PE, router and queues at (i, j) location, $T_{ij,PE}$, $T_{ij,r}$ and $T_{ij,k,q}$ are chip temperatures of the PE, router and queues at (i, j) location, $S_{ij,PE}$, $S_{ij,r}$ and $S_{ij,k,q}$ are the subthreshold swing parameters of the PE, router and queues at (i, j) location, and q_e is the electron charge.

Although we consider in Eq. 6.27 the static power consumption as a function of the supply and threshold voltages, to simplify the mathematical derivations of the optimal controller we use the alpha-power law model in [135] and express all the threshold voltage dependencies in the cost function and constraints in terms of the supply voltage and operating frequency of both PEs and routers. In addition, instead of using the relationship between the threshold voltage and the circuit temperature, in what follows, we determine the dependency between the peak temperature of NoC components and the operating frequency. Nevertheless, the current power and thermal management formulation can be extended for time dependent temperature profiles.

The peak temperature (PT) terms introduced in Eq. 6.13 are defined by the following relations:

$$\begin{aligned} PT_{ij,PE} &= B_{ij,PE} f_{ij,PE}^{\epsilon_{ij,PE}} \\ PT_{ij,r} &= B_{ij,r} f_{ij}^{\epsilon_{ij,r}} \\ PT_{ij,k,q} &= B_{ij,k,q} f_{ij}^{\epsilon_{ij,k,q}} \end{aligned} \quad (6.28)$$

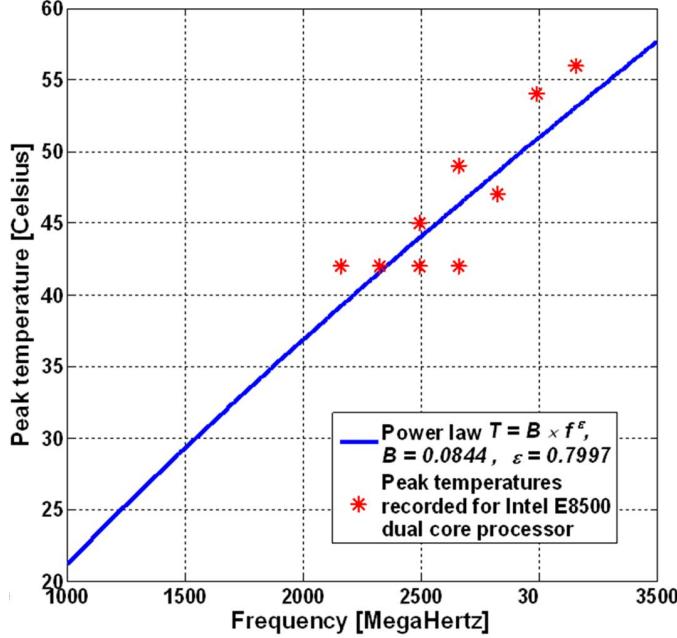


Figure 6.8: Peak temperature displays a power law dependency (with exponent $\epsilon = 0.7997$) on the operating frequency for Intel E8500 dual core processor confirming the scaling law in Eq. 6.28.

where $B_{ij,PE}$, $B_{ij,r}$ and $B_{ij,k,q}$ are the proportionality constants between the peak temperature and operating frequency of the PE, router and queues at (i,j) location, $\epsilon_{ij,PE}$, $\epsilon_{ij,r}$ and $\epsilon_{ij,k,q}$ are the power law exponents of the dependency between peak temperature profile and the operating frequency of the PE, router and queues at (i,j) location.

To validate the non-linear relationship between the peak temperature profile $PT_{ij,PE}$ and the operating frequency $f_{ij,PE}$ in Eq. 6.28 we measure the temperature profile of an Intel quad-core i7 920 (see Figure 6.8) as a function of several operating frequencies. By using the least-square method, we observe that the empirical temperatures are better fitted by a power law $PT \approx f^{0.79}$. Note that, one can relax the nonlinear relationship between temperature and frequency and consider a linear dependence, thus obtaining an upper bound on temperature profile.

To avoid the situation in which the optimal controller selects an unattainable control signal (either very small or very large frequency), the optimization problem is augmented with the constraints on the operating frequency for both PEs and routers:

$$\begin{aligned}
V_{ij,PE}^{min}(t) &\leq V_{ij,PE}(t) \leq V_{ij,PE}^{max}(t) \\
V_{ij}^{min}(t) &\leq V_{ij}(t) \leq V_{ij}^{max}(t), \quad i = 1 \div M, \quad j = 1 \div N \\
f_{ij,PE}^{min}(t) &\leq f_{ij,PE}(t) \leq f_{ij,PE}^{max}(t) \\
f_{ij}^{min}(t) &\leq f_{ij}(t) \leq f_{ij}^{max}(t), \quad i = 1 \div M, \quad j = 1 \div N
\end{aligned} \tag{6.29}$$

Consequently, $V_{ij,PE}^{min}$ and $V_{ij,PE}^{max}$ are the time-dependent lower and upper bounds on the supply voltage of the PE at (i,j) location; $f_{ij,PE}^{min}$ and $f_{ij,PE}^{max}$ are the time-dependent lower and upper bounds

on the operating frequency of the PE at (i, j) location; V_{ij}^{min} and V_{ij}^{max} are the lower and upper bounds on the router supply voltage at (i, j) location on the mesh; f_{ij}^{min} and f_{ij}^{max} are the lower and upper bounds on the router operating frequency at (i, j) location on the mesh. Besides capturing the application dependent performance constraints, the role of considering these time-dependent lower and upper bounds on both supply voltage and operating frequency is to account for circuit aging (*e.g.*, negative bias temperature instability, time dielectric breakdown, gate oxide integrity, hot carrier injection, electromigration) and system non-idealities (*e.g.*, parameter variations [114], short channel effect [156]).

In order to solve the above optimization problem we use concepts from calculus of variations [7][158] and optimization theory by constructing first the Lagrangian functional as follows:

$$\begin{aligned}
L(y_{ij}, V_{ij,PE}, f_{ij,PE}, \lambda_{ij,PE}, \beta_{ij,PE}, \eta_{ij,PE}, V_{ij}, f_{ij}, x_{ij}^k, \gamma_{ij,k,q}) = & \int_{t_i}^{t_f} \sum_{i,j=1}^{M,N} \left[\frac{q_{ij}}{2} \left(y_{ij}(t) - y_{ij}^{ref}(t) \right)^2 + \right. \\
& + \frac{r_{ij,PE} f_{ij,PE}^2(t)}{2} + C_L^{PE} V_{ij,PE}^2 f_{ij,PE} S w_{ij,PE} + B_{ij,PE} f_{ij,PE}^{\epsilon_{ij,PE}} + C_L^r V_{ij}^2 f_{ij} S w_{ij,r} + \\
& + \frac{(1+m_{ij,PE}) K_{ij,PE} (k_B T_{ij,PE})^2}{q_e^2} V_{ij,PE} e^{-\frac{V_{ij,PE} - (G_{ij,PE} f_{ij,PE} V_{ij,PE})}{S_{ij,PE} \delta_{ij,PE}^{\frac{1}{\delta_{ij,PE}}}}} + B_{ij,r} f_{ij}^{\epsilon_{ij,r}} + \\
& + \lambda_{ij,PE} \left(\frac{d^{\alpha_{ij}^{PE}} y_{ij}(t)}{dt^{\alpha_{ij}^{PE}}} - a_{ij}^{PE}(t) y_{ij}(t) - b_{ij}^{PE}(t) f_{ij,PE}(t) + c_{ij}^{PE}(t) f_{ij}(t) \right) + \quad (6.30) \\
& + \frac{r_{ij} f_{ij}^2(t)}{2} + \frac{(1+m_{ij,r}) K_{ij,r} (k_B T_{ij,r})^2}{q_e^2} V_{ij} e^{-\frac{V_{ij} - (G_{ij} f_{ij} V_{ij})}{S_{ij,r} \delta_{ij,r}^{\frac{1}{\delta_{ij,r}}}}} + \sum_{k=0}^4 [w_{ij,k,q} \left(\frac{q_{ij}^k}{2} \left(x_{ij}^k(t) - x_{ij}^{k,ref} \right)^2 \right. \\
& \left. + C_L^q V_{ij}^2 f_{ij} S w_{ij,k,q} + \frac{(1+m_{ij,k,q}) K_{ij,k,q} (k_B T_{ij,k,q})^2}{q_e^2} V_{ij} e^{-\frac{V_{ij} - (G_{ij} f_{ij} V_{ij})}{S_{ij,k,q} \delta_{ij,q}^{\frac{1}{\delta_{ij,q}}}}} + B_{ij,k,q} f_{ij}^{\epsilon_{ij,k,q}} + \right. \\
& \left. \gamma_{ij,k,q} \left(\frac{d^{\alpha_{ij}^k} x_{ij}^k(t)}{dt^{\alpha_{ij}^k}} - a_{ij}^k(t) x_{ij}^k(t) + b_{ij}^k(t) f_{ij,neig}(t) + c_{ij}^k(t) f_{ij}(t) \right) \right)] dt
\end{aligned}$$

where y_{ij} and $x_{ij,k,q}$ denote the queue utilization variables, V_{ij} , $V_{ij,PE}$, f_{ij} , and $f_{ij,PE}$ are the supply voltage and frequency associated with the router and PE at (i, j) location, $\lambda_{ij,PE}$, $\beta_{ij,PE}$ and $\eta_{ij,PE}$ are the Lagrange multipliers associated with the constraints imposed for the queue utilization and the operating frequency of the PE at (i, j) location, $\delta_{ij,PE}$, $\delta_{ij,r}$ and $\delta_{ij,q}$ are the technology dependent parameters of the PE, router and queues at (i, j) location, $\epsilon_{ij,PE}$, $\epsilon_{ij,r}$ and $\epsilon_{ij,k,q}$ are the power law exponents of the dependency between peak temperature and the operating frequency of the PE, queue and j -th router at (i, j) location, and $\gamma_{ij,k,q}$ are the Lagrange multipliers associated with the constraints imposed on the utilization of the queues between neighboring routers in different VFIs.

In addition to the technological constraints such as the operating frequency which is bounded between a minimum and maximum frequency, we also have some boundary constraints for the

utilization of the mixed clock queues as follows:

$$\begin{aligned} y_{ij}(t_i) &= y_{ij}^0 \quad y_{ij}(t_f) = y_{ij}^{ref} \quad i = 1 \div M, j = 1 \div N \\ x_{ij}^k(t_i) &= x_{ij}^{k,0} \quad x_{ij}^k(t_f) = x_{ij}^{k,ref}, \quad k = 0, 1, 2, 3, 4 \end{aligned} \quad (6.31)$$

which are required by the necessity of satisfying certain computational performance levels by the end of the control interval $[t_i, t_f]$. Note that, in Eq. 6.31, y_{ij}^0 and y_{ij}^{ref} represent the initial and final reference utilization values for the queue between the PE and the router at (i, j) location, $x_{ij}^{k,0}$ and $x_{ij}^{k,ref}$ denote the initial and final reference utilization values for the queues associated with the router at (i, j) location.

Following the calculus of variations methodology, we assume that the queue utilization functionals satisfy the following relations:

$$\begin{aligned} y_{ij}(t) &= y_{ij}^{optim}(t) + \tau\phi_{ij}(t) \\ x_{ij}^k(t) &= x_{ij}^{k,optim}(t) + \tau\varphi_{ij}(t) \end{aligned} \quad (6.32)$$

where $y_{ij}^{optim}(t)$ and $x_{ij}^{k,optim}(t)$ are queue utilizations at which the Lagrange function in Eq. 6.30 attains its minimum, τ is an infinitesimal perturbation factor, $\phi_{ij}(t)$ and $\varphi_{ij}(t)$ are variational functions vanishing at time t_f , i.e., $\phi_{ij}(t_f) = \varphi_{ij}(t_f) = 0$.

By expanding the Lagrange functional via the Taylor formula and considering that it attains its extremum (minimum) in the vicinity of $\tau = 0$, i.e., $\partial L/\partial\tau = 0$, we obtain the following relations:

$$\begin{aligned} \frac{\partial L}{\partial y_{ij}} + {}_t D_{t_f}^{\alpha_{ij}^{PE}} \frac{\partial L}{\partial {}_{t_i} D_t^{\alpha_{ij}^{PE}} y_{ij}} &= 0, \quad \frac{\partial L}{\partial f_{ij,PE}} = 0, \quad \frac{\partial L}{\partial V_{ij,PE}} = 0, \\ \frac{\partial L}{\partial \lambda_{ij,PE}} &= 0, \quad i = 1 \div M, \\ \frac{\partial L}{\partial x_{ij}^k} + {}_t D_{t_f}^{\alpha_{ij}^k} \frac{\partial L}{\partial {}_{t_i} D_t^{\alpha_{ij}^k} x_{ij}^k} &= 0, \quad \frac{\partial L}{\partial V_{ij}} = 0, \quad \frac{\partial L}{\partial f_{ij}} = 0, \\ \frac{\partial L}{\partial \gamma_{ij,k,q}} &= 0, \quad j = 1 \div N, \quad k = 0 \div 4 \end{aligned} \quad (6.33)$$

Since Eq. 6.33 are of Euler-Lagrange type of equations and need to be satisfied for all functions $\phi_{ij}(t)$ and $\varphi_{ij}(t)$, it implies that the optimal queue utilizations are given by the following relations:

$$r_{ij} \left(y_{ij} - y_{ij}^{ref} \right) + a_{ij}^{PE} \lambda_{ij,PE} - {}_t D_{t_f}^{\alpha_{ij}^{PE}} \lambda_{ij,PE} = 0 \quad (6.34)$$

$$\begin{aligned}
& f_{ij,PE} \{ 1.55^{\epsilon_{ij,PE}-2} B_{ij,PE} \epsilon_{ij,PE} (\epsilon_{ij,PE} - 1) + r_{ij,PE} + \frac{z_{ij,PE} u_{ij,PE}}{2.4025 \delta_{ij,PE}} [0.4 \frac{u_{ij,PE}}{\delta_{ij,PE}} - \frac{1}{\delta_{ij,PE} S_{ij,PE}} \\
& + \frac{1}{S_{ij,PE}} + \frac{1}{\delta_{ij,PE}} - 1 - u_{ij,PE}] \} + V_{ij,PE} \{ 2C_L^{PE} S_{w_{ij,PE}} + \frac{z_{ij,PE} u_{ij,PE}}{2.4025 \delta_{ij,PE}} [-1.24 \frac{u_{ij,PE}}{\delta_{ij,PE}^3} - \\
& - 0.31 \frac{u_{ij,PE}}{\delta_{ij,PE}^2} + 3.41 \frac{u_{ij,PE}}{\delta_{ij,PE}} - \frac{1.55}{\delta_{ij,PE} S_{ij,PE}} + \frac{7.75}{\delta_{ij,PE}} - \frac{3.1}{S_{ij,PE}} - 0.93 u_{ij,PE} - 4.65] \} + \\
& + \frac{z_{ij,PE} u_{ij,PE}}{2.4025 \delta_{ij,PE}} [\frac{1.24 u_{ij,PE}}{\delta_{ij,PE}^2} - \frac{2.48 u_{ij,PE}}{\delta_{ij,PE}} - \frac{3.1}{\delta_{ij,PE} S_{ij,PE}} - \frac{3.1}{\delta_{ij,PE}} + \frac{6.2}{S_{ij,PE}} + 0.31 u_{ij,PE} + 1.55] - \\
& - C_L^{PE} S_{w_{ij,PE}} + 1.55^{\epsilon_{ij,PE}-1} \epsilon_{ij,PE} (2 - \epsilon_{ij,PE}) B_{ij,PE} + \lambda_{ij,PE} b_{ij,PE} - c_{ij}^0 \gamma_{ij,0,q} = 0 \quad (6.35)
\end{aligned}$$

$$\begin{aligned}
& V_{ij,PE} \left\{ z_{ij,PE} \left[\frac{2}{S_{ij,PE}} - \frac{2u_{ij,PE}}{\delta_{ij,PE}} - u_{ij,PE} - 6 \right] + \frac{0.4 u_{ij,PE}^2}{\delta_{ij,PE}^2} \right\} + \\
& + f_{ij,PE} \left\{ z_{ij,PE} \left[\frac{u_{ij,PE}}{\delta_{ij,PE}} \left(\frac{2u_{ij,PE} - \frac{1}{S_{ij,PE}} + 1}{7.69 \delta_{ij,PE}} - \frac{0.13}{S_{ij,PE}} + 0.13 \right) - 0.13 \right] + C_L^{PE} S_{w_{ij,PE}} \right\} - \\
& - 3.1 C_L^{PE} S_{w_{ij,PE}} + z_{ij,PE} \left[\frac{u_{ij,PE}}{\delta_{ij,PE}} \left(0.6 - \frac{2}{\delta_{ij,PE}} \right) - \frac{0.2}{S_{ij,PE}^2} + 0.2 u_{ij,PE} + 1 \right] = 0 \quad (6.36)
\end{aligned}$$

$$\frac{d^{\alpha_{ij}^{PE}} y_{ij}(t)}{dt^{\alpha_{ij}^{PE}}} = a_{ij}^{PE}(t) y_{ij}(t) + b_{ij}^{PE}(t) f_{ij,PE}(t) - c_{ij}^{PE}(t) f_{ij}(t) \quad (6.37)$$

$$q_{ij}^k \left(x_{ij}^k - x_{ij}^{k,ref} \right) + a_{ij}^k \gamma_{ij,k,q} - {}_t D_{t_f}^{\alpha_{ij}^k} \gamma_{ij,k,q} = 0 \quad (6.38)$$

$$\frac{d^{\alpha_{ij}^k} x_{ij}^k(t)}{dt^{\alpha_{ij}^k}} - a_{ij}^k(t) x_{ij}^k(t) + b_{ij}^k(t) f_{ij,neig}(t) + c_{ij}^k(t) f_{ij}(t) = 0 \quad (6.39)$$

$$\begin{aligned}
& f_{ij} \left\{ r_{ij} + A_{ij}^r + A_{ij}^q \right\} + V_{ij} \left\{ 2C_L^r S_{w_{ij,r}} + \frac{z_{ij,r} u_{ij,r}}{2.4025 \delta_{ij}} F_{ij}^r + \sum_{k=0}^4 \frac{w_{ij,k,q} z_{ij,k,q} u_{ij,k,q}}{2.4025 \delta_{ij}} F_{ij}^q \right\} + \\
& + Q_{ij}^r + Q_{ij}^q + Q_{ij} - C_L^r S_{w_{ij,r}} - \sum_{k=0}^4 2w_{ij,k,q} C_L^q S_{w_{ij,k,q}} + w_{i-1,j,1,q} b_{i-1,j}^1 \gamma_{i-1,j,1,q} + \\
& + w_{i+1,j,2,q} b_{i+1,j}^2 \gamma_{i+1,j,2,q} + w_{ij-1,3,q} b_{ij-1}^3 \gamma_{ij-1,3,q} + w_{ij+1,4,q} b_{ij+1}^4 \gamma_{ij+1,4,q} + w_{ij,0,q} b_{ij}^0 \gamma_{ij,0,q} \\
& - \sum_{k=0}^4 w_{ij,k,q} c_{ij}^k \gamma_{ij,k,q} - c_{ij}^{PE} \lambda_{ij,PE} = 0 \quad (6.40)
\end{aligned}$$

$$\begin{aligned}
& V_{ij} \left\{ W_{ij}^r + \sum_{k=0}^4 w_{ij,k,q} W_{ij,k}^q \right\} + f_{ij} \left\{ X_{ij}^r + \sum_{k=0}^4 \left[w_{ij,k,q} z_{ij,k,q} X_{ij,k}^q + C_L^q S w_{ij,k,q} \right] \right\} \\
& - 3.1 C_L^r S w_{ij,r} - \sum_{k=0}^4 3.1 w_{ij,k,q} C_L^q S w_{ij,k,q} + z_{ij,r} \left[\frac{u_{ij,r}}{\delta_{ij}} \left(0.6 - \frac{2}{\delta_{ij}} \right) - \frac{0.2}{S_{ij,r}^2} + 0.2 u_{ij,r} + 1 \right] \\
& + \sum_{k=0}^4 w_{ij,k,q} z_{ij,k,q} \left[\frac{u_{ij,k,q}}{\delta_{ij}} \left(0.6 - \frac{2}{\delta_{ij}} \right) - \frac{0.2}{S_{ij,k,q}^2} + 0.2 u_{ij,k,q} + 1 \right] = 0
\end{aligned} \tag{6.41}$$

where we used the following Taylor (expanssion) approximations:

$$e^{-V_{ij,PE}^{th}/S_{ij,PE}} \approx 1 - \frac{\left(V_{ij,PE} - (K_{ij,PE} V_{ij,PE} f_{ij,PE})^{\frac{1}{\delta_{ij,PE}}} \right)}{5 S_{ij,PE}} \tag{6.42}$$

$$e^{-V_{ij}^{th}/S_{ij,r}} \approx 1 - \frac{\left(V_{ij} - (K_{ij,r} V_{ij} f_{ij})^{\frac{1}{\delta_{ij}}} \right)}{5 S_{ij,r}} \tag{6.43}$$

$$f_{ij,PE}^2 \approx 3.1 f_{ij,PE} - 2.4025 \text{ for } f_{ij,PE} \in [0.1 GHz, 4 GHz] \tag{6.44}$$

$$f_{ij}^2 \approx 3.1 f_{ij} - 2.4025 \text{ for } f_{ij} \in [0.1 GHz, 4 GHz] \tag{6.45}$$

$$f_{ij,PE}^{\epsilon_{ij,PE}-1} \approx 1.55^{\epsilon_{ij,PE}-2} (\epsilon_{ij,PE} - 1) f_{ij,PE} + 1.55^{\epsilon_{ij,PE}-1} (2 - \epsilon_{ij,PE}) \\ \text{for } f_{ij,PE} \in [0.1 GHz, 4 GHz], \epsilon_{ij,PE} \in [1.1, 2.5] \tag{6.46}$$

$$f_{ij}^{\epsilon_{ij}-1} \approx 1.55^{\epsilon_{ij}-2} (\epsilon_{ij} - 1) f_{ij} + 1.55^{\epsilon_{ij}-1} (2 - \epsilon_{ij}) \\ \text{for } f_{ij} \in [0.1 GHz, 4 GHz], \epsilon_{ij} \in [1.1, 2.5] \tag{6.47}$$

$$V_{ij,PE}^2 \approx 2 V_{ij,PE} - 1 \text{ for } V_{ij,PE} \in [0.4V, 1.4V] \tag{6.48}$$

$$V_{ij}^2 \approx 2 V_{ij} - 1 \text{ for } V_{ij} \in [0.4V, 1.4V] \tag{6.49}$$

$$f_{ij,PE} V_{ij,PE} \approx 1.55 V_{ij,PE} + f_{ij,PE} - 1.55 \tag{6.50}$$

$$f_{ij,PE} V_{ij} \approx 1.55 V_{ij} + f_{ij} - 1.55 \tag{6.51}$$

and the parameters $z_{ij,PE}$, $u_{ij,PE}$, $z_{ij,r}$, $u_{ij,r}$, $z_{ij,k,q}$, $u_{ij,k,q}$, A_{ij}^r , A_{ij}^q , $Z_{ij}^{k,q}$, F_{ij}^r , F_{ij}^q , Q_{ij}^r , Q_{ij}^q , Q_{ij} , W_{ij}^r , $W_{ij,k}^q$, X_{ij}^r and $X_{ij,k}^q$ are given by the following relations:

$$z_{ij,PE} = \frac{(1 + m_{ij,PE}) K_{ij,PE} k_B^2 T_{ij,PE}^2}{q_e^2} \tag{6.52}$$

$$u_{ij,PE} = \frac{(1.55 * G_{ij,PE})^{(1/\delta_{ij,PE})}}{S_{ij,PE}} \tag{6.53}$$

$$z_{ij,r} = \frac{(1 + m_{ij,r}) K_{ij,r} k_B^2 T_{ij,r}^2}{q_e^2} \tag{6.54}$$

$$u_{ij,r} = \frac{(1.55 * G_{ij,r})^{(1/\delta_{ij,r})}}{S_{ij,r}} \quad (6.55)$$

$$z_{ij,k,q} = \frac{(1 + m_{ij,k,q}) K_{ij,k,q} k_B^2 T_{ij,k,q}^2}{q_e^2} \quad (6.56)$$

$$u_{ij,k,q} = \frac{(1.55 * G_{ij,k,q})^{(1/\delta_{ij,q})}}{S_{ij,k,q}} \quad (6.57)$$

$$A_{ij}^r = \frac{1.55^{\epsilon_{ij,r}} B_{ij,r} \epsilon_{ij,r} (\epsilon_{ij,r} - 1)}{2.4025} + \frac{z_{ij,r} u_{ij,r}}{2.4025 \delta_{ij}} \left(0.4 \frac{u_{ij,r}}{\delta_{ij}} - \frac{1}{\delta_{ij} S_{ij,r}} + \frac{1}{\delta_{ij}} + \frac{1}{S_{ij,r}} - 0.2 u_{ij,r} - 1 \right) \quad (6.58)$$

$$A_{ij}^q = \sum_{k=0}^4 \frac{w_{ij,k,q}}{2.4025} \left\{ Z_{ij}^{k,q} + \frac{z_{ij,k,q} u_{ij,k,q}}{\delta_{ij}} \left(\frac{1}{S_{ij,k,q}} - \frac{0.4 u_{ij,k,q} - \frac{1}{S_{ij,k,q} + 1}}{\delta_{ij}} (0.2 u_{ij,k,q} + 1) \right) \right\} \quad (6.59)$$

$$Z_{ij}^{k,q} = 1.55^{\epsilon_{ij,k,q}} B_{ij,k,q} \epsilon_{ij,k,q} (\epsilon_{ij,k,q} - 1) \quad (6.60)$$

$$F_{ij}^r = \frac{7.75 - \frac{1.55}{S_{ij,r}} - \frac{1.24 u_{ij,r}}{\delta_{ij}^2} - 0.31 \frac{u_{ij,r}}{\delta_{ij}} + 3.41 u_{ij,r}}{\delta_{ij}} - 4.65 - 0.93 u_{ij,r} - \frac{1}{S_{ij,r}} \quad (6.61)$$

$$F_{ij}^q = \frac{3.41 u_{ij,k,q}}{\delta_{ij}} - 1.24 \frac{u_{ij,k,q}}{\delta_{ij}^3} - 0.31 \frac{u_{ij,k,q}}{\delta_{ij}^2} + \frac{7.75}{\delta_{ij}} - \frac{1.55}{\delta_{ij} S_{ij,k,q}} - \frac{3.1}{S_{ij,k,q}} - 4.65 - 0.93 u_{ij,k,q} \quad (6.62)$$

$$Q_{ij}^r = \frac{z_{ij,r} u_{ij,r}}{2.4025 \delta_{ij}} \left[\frac{1.24 u_{ij,r}}{\delta_{ij}^2} - \frac{3.1}{\delta_{ij}} - \frac{2.48 u_{ij,r}}{\delta_{ij}} - \frac{3.1}{\delta_{ij} S_{ij,r}} + \frac{6.2}{S_{ij,r}} + 0.31 u_{ij,r} + 1.55 \right] \quad (6.63)$$

$$Q_{ij}^q = \frac{w_{ij,k,q} z_{ij,k,q} u_{ij,k,q}}{2.4025 \delta_{ij}} \left[\frac{\frac{1.24 u_{ij,k,q}}{\delta_{ij}} - 2.48 u_{ij,k,q} - \frac{3.1}{S_{ij,k,q}} - 3.1}{\delta_{ij}} + \frac{6.2}{S_{ij,k,q}} + 0.31 u_{ij,k,q} + 1.55 \right] \quad (6.64)$$

$$Q_{ij} = \frac{1.55^{\epsilon_{ij,r}} \epsilon_{ij,r} (2 - \epsilon_{ij,r}) B_{ij,r}}{1.55} + \sum_{k=0}^4 \frac{1.55^{\epsilon_{ij,k,q}} w_{ij,k,q} \epsilon_{ij,k,q} (2 - \epsilon_{ij,k,q}) B_{ij,k,q}}{1.55} \quad (6.65)$$

$$W_{ij}^r = z_{ij,r} \left[\frac{\frac{2}{S_{ij,r}} - \frac{2u_{ij,r}}{\delta_{ij}} - \frac{u_{ij,r}}{\delta_{ij}^2} - u_{ij,r} - 6}{5S_{ij,r}} + \frac{0.4u_{ij,r}^2}{\delta_{ij}^2} \right] + 3.1C_L^r S w_{ij,r} \quad (6.66)$$

$$W_{ij,k}^q = z_{ij,k,q} \left[\frac{\frac{2}{S_{ij,k,q}} - \frac{2u_{ij,k,q}}{\delta_{ij}} - \frac{u_{ij,k,q}}{\delta_{ij}^2} - u_{ij,k,q} - 6}{5S_{ij,k,q}} + \frac{0.4u_{ij,k,q}^2}{\delta_{ij}^2} \right] + 3.1C_L^q S w_{ij,k,q} \quad (6.67)$$

$$X_{ij}^r = z_{ij,r} \left[\frac{u_{ij,r}}{\delta_{ij}} \left(\frac{2u_{ij,r} - \frac{1}{S_{ij,r}} + 1}{7.69\delta_{ij}} - \frac{0.13}{S_{ij,r}} + 0.13 \right) - 0.13 \right] + C_L^r S w_{ij,r} \quad (6.68)$$

$$X_{ij,k}^q = \frac{u_{ij,k,q}}{\delta_{ij}} \left[\frac{2u_{ij,k,q} - \frac{1}{S_{ij,k,q}} + 1}{7.69\delta_{ij}} - \frac{0.13}{S_{ij,k,q}} + 0.13 \right] - 0.13 \quad (6.69)$$

Note that more details about the derivations of these equations and the meaning of all the parameters introduced above can be found in [35].

In order to determine the optimal supply voltage and frequency (implicitly threshold voltage) values for which the queue utilization references are followed, we divide the time domain $[t_i, t_f]$ into $N = [t_f - t_i]/\Delta t$ equal domains, where Δt is the discretization interval. Under this discretization strategy and using the Grunwald-Letnikov formula from Eq. 1.4 for expressing the fractional differentiation, the optimal values are determined by solving a linear system made of the relations in Eq. 6.34, Eq. 6.35, Eq. 6.36, Eq. 6.37, Eq. 6.38 and Eq. 6.39.

6.5 Experimental results

To evaluate the impact of the fractal optimal control approach to power and temperature management, we consider a combination of trace driven and cycle accurate simulation of a VFI-based NoC architecture. From an application perspective, we consider five 16-node multithreaded commercial workloads (see Table 6.1) obtained by running them on a FLEXUS based shared-memory 16-processor environment consisting of cycle accurate models of out-of-order processors and cache hierarchy [160]. More precisely, we evaluate the performance of the control strategy for an Apache HTTP server v2.0 from SPECweb99 benchmark [146]. The online transaction processing applications consist of TPC-C v3.0 workload on both IBM DB2 v8 ESE and Oracle 10gEnterprise Database Server. The scientific applications consist of an ocean simulation and a blocked sparse Cholesky factorization.

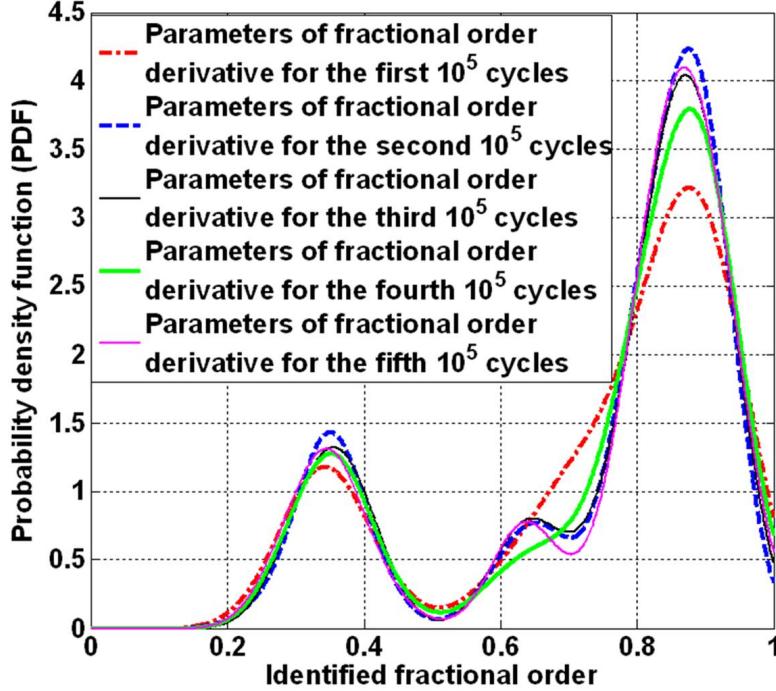


Figure 6.9: Distribution of the identified parameters corresponding to the fractional order derivatives characterizing the queue utilization dynamics (see Eq. 6.14) for a 4×4 mesh NoC employing XY wormhole routing with input queue sizes of 15 flits and running a 16 node multithreaded web server (apache HTTP server v2.0) application. There is a slight variation in the distribution of fractional orders of the derivative as a function of the interval of the running application. This justifies the need for dynamically identifying the parameters of the model and then solving the same optimization problem with different coefficients as a function of the application phases for better power management.

From an architecture perspective, we consider a 4×4 mesh NoC employing XY and adaptive wormhole routing schemes and two configurations: a) mixed clock queues of 15 and 30 flits in size and packets consisting of 10 and 25 flits, respectively. The execution of each application is divided into several intervals (*e.g.*, these intervals can be $1\mu s$ long). For an interval of $10\mu s$, the system identification module is called to estimate from the arrival, departure and queue utilization processes the parameters that fit best in the least square sense the first $10\mu s$ of the last $11\mu s$. After the identification step is completed in parallel with the application computations, the fractal optimal controller reads the estimated parameters and solves the optimization problem defined by Eq. 6.34, Eq. 6.35, Eq. 6.36, Eq. 6.37, Eq. 6.38 and Eq. 6.39 to determine the operating frequencies and supply voltages that ensure a predefined performance level specified in terms of queue utilization references.

In order to validate our dynamic optimization procedure in the presence of fractal characteristics, we consider a 4×4 mesh NoC employing XY wormhole routing with input queue sizes of 15 flits and running a 16 node multithreaded web server (apache HTTP server v2.0) application. We first call the parameter identification procedure which estimates both the fractional orders of the

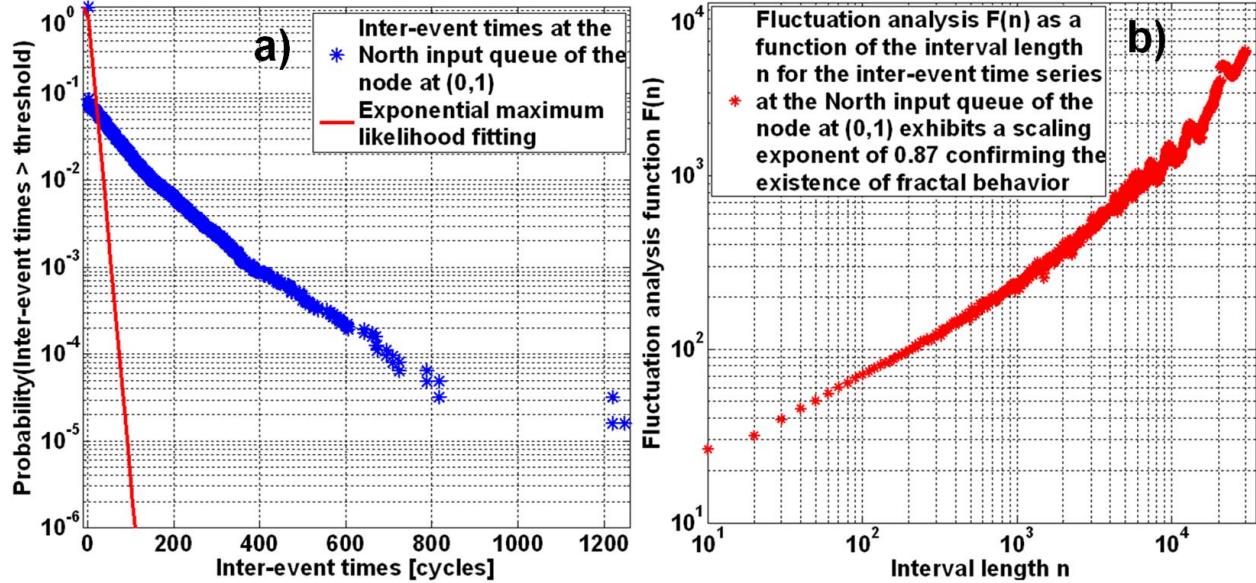


Figure 6.10: a) Probability of the inter-events times between successive changes in queue utilization at north input queue of the node at (0,1) to exceed a given threshold cannot be approximated by an exponential distribution. These inter-event times were obtained for a 4×4 mesh NoC employing XY wormhole routing with input queue sizes of 15 flits and running a 16 node multithreaded web server (apache HTTP server v2.0) application. b) Detrended fluctuation analysis of the inter-event times between successive changes in queue utilization at north input queue of the node at (0,1) reveals a scaling exponent of 0.87 and confirms the exitence of long-range dependence and fractal behavior in the dynamics of the queue utilization process.

derivatives and the queue utilization, writing and reading coefficients characterizing the queue utilization process of each queue. Figure 6.9 shows the distribution of the identified parameters of the fractional order derivatives characterizing the dynamics of the queue utilization process across all queues in a 4×4 mesh NoC. One can note that the fractional orders are concentrated around 0.38 and 0.82, respectively. Moreover, there is a slight variation in the distribution of fractional orders as a function of the application phase.

To further investigate the need for adopting a fractal characterization, and thus, a fractional order derivative model, we have also investigated the behavior of the probability of the inter-event times between consecutive changes in the queue utilization at the north input queue for the node at (0,1) to exceed a given threshold. From Figure 6.10.a, we can notice that this probability cannot be fitted by an exponential distribution.

For completeness, we have also performed the detrended fluctuation analysis of the inter-event times series between successive changes in the queue utilization. Detrended fluctuation analysis is a method used to quantify the fractal behavior in non-stationary stochastic processes [124] by estimating the scaling exponent α of the fluctuation analysis functional $F(n)$ as a function of the interval length n , *i.e.*, $F(n) \approx n^\alpha$. If the estimated scaling exponent α is larger than 0.5, then the stochastic process is considered to posses fractal characteristics.

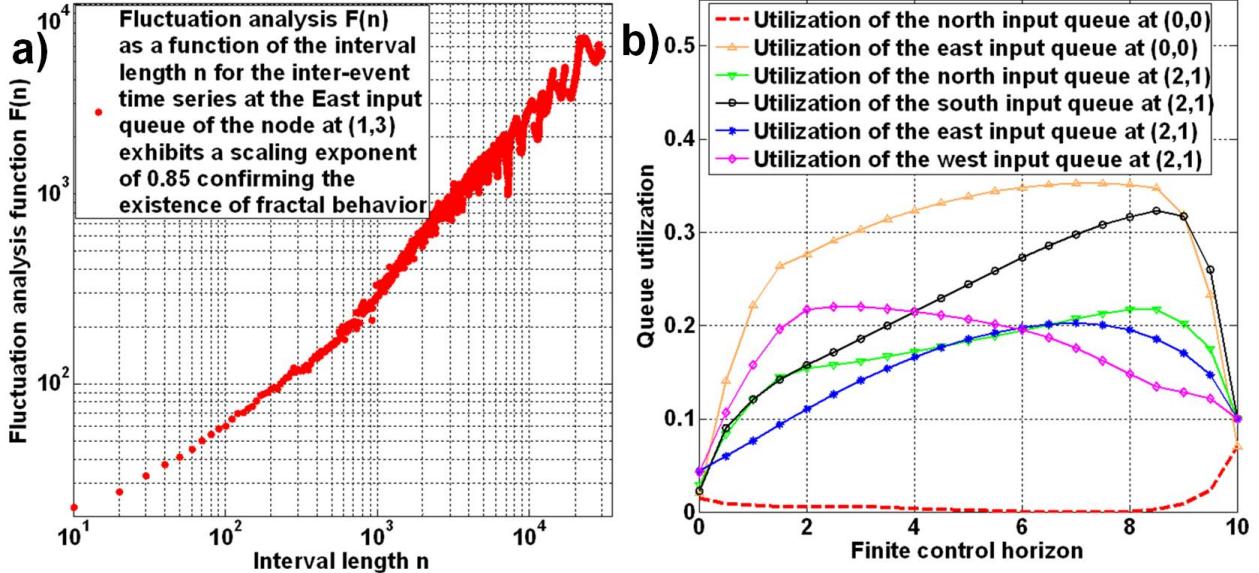


Figure 6.11: a) Detrended fluctuation analysis of the inter-event times between successive changes in queue utilization at east input queue of the node at (1,3) reveals a scaling exponent of 0.85 and confirms the existence of fractal behavior in the dynamics of the queue utilization process. b) Computed queue utilization for six queues at tiles (0,0) and (2,1) and a finite control horizon of 1000 cycles. The controllers determine the operating supply voltages and frequencies for each router in the network such that the queues of the nodes at (1,1), (1,2), (2,1) and (2,2) reach a queue utilization reference value of 0.1 and the queues of the remaining nodes reach a 0.07 reference value. The proposed optimal controller is able to stabilize the queues at (0,0) and (2,1) at a reference value of 0.07 and 0.1, respectively.

Figure 6.10.b and Figure 6.11.a show the fluctuation analysis $F(n)$ as a function of the interval length n for the inter-event times between consecutive changes in the queue utilization at the north input queue for the node at (0,1) and the east input queue for the node at (1,3), respectively. Both graphs exhibit scaling exponents larger than 0.5 confirming the existence of fractal behavior in the queue utilization processes, and thus justifying the need for adopting a fractional calculus approach to the description of the queue dynamics as shown in Eq. 6.14 through Eq. 6.24.

By estimating the parameters of the model via least square method and solving the linear system in Eq. 6.34, Eq. 6.35, Eq. 6.36, Eq. 6.37, Eq. 6.38 and Eq. 6.39 for 20 discrete steps we obtain the operating supply voltage and frequency of each tile necessary to reach the predefined reference values for queue utilization.

As shown in [37], decreasing the minimum supply voltage is one of the most efficient technique to ensure lower energy consumption in high-performance multicore microprocessors. Consequently, we constrain the optimal controller to choose supply voltages between 0.6V and 1.4V and frequencies between 0.1GHz and 3GHz, respectively. Regarding the queue utilization references, we constrain the utilizations of the queues at nodes (1,1), (1,2), (2,1), and (2,2) to be brought to a 0.1 reference value and the queues in the remaining nodes to attain a 0.07 reference value.

Figure 6.11.b shows the computed queue utilization of the input queues at tiles (0,0) and (2,1),

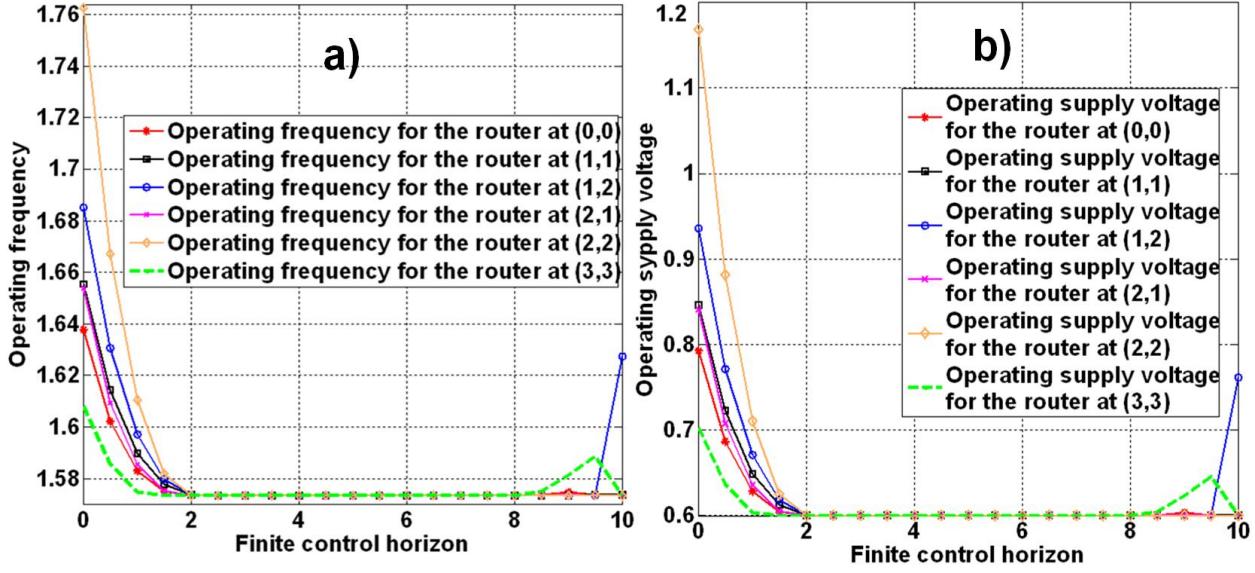


Figure 6.12: a) The variation of operating frequencies for the routers at (0,0), (1,1), (1,2), (2,1), (2,2) and (3,3) necessary to reach the reference values imposed on queues utilization b) The necessary supply voltage for the routers at (0,0), (1,1), (1,2), (2,1), (2,2) and (3,3) to meet the imposed queue utilization references.

respectively. We can observe that the optimal controller is able to bring the utilization of the queues in tiles (0,0) and (2,1) at the reference value of 0.07 and 0.1, respectively. Figure 6.12.a and Figure 6.12.b show the operating frequencies and supply voltages of the nodes at (0,0), (1,1), (1,2), (2,2), and (3,3) needed to attain the imposed reference values. By comparing the power consumption of the new VFI system with an NoC architecture with all PEs and routers running at 3GHz and 1V, we obtain approximately 60% power savings.

In this chapter, we have addressed the problem of power and thermal management in VFI-based NoC architectures where computational workloads are highly complex and exhibit fractal characteristics. To overcome the limitations of short-range memory models used in classical linear system theory, we have proposed a new modeling approach based on the dynamics of queue utilization via fractional differential equations. This fractal model is used to formulate an optimal control problem, which determines the necessary operating frequencies and supply voltages so that the NoC queues reach their target reference values. Nevertheless, the fractal dynamic optimization introduced in this chapter can easily be extended to other performance metrics (e.g., clock cycles per instruction, stall times, source-to-destination latency).

Chapter 7

Modeling, Analysis and Optimization of Cyber-Physical Systems

Cyber-physical systems (CPS) represent the information technology quest of the 21-st century for a better, cleaner, safer life which integrates computation, communication, and control with physical processes. Physical processes are predominantly non-stationary and require time-dependent models for modeling and understanding their behavior. In contrast, in most current computing platforms, their workloads and design methodologies lack proper models for the time component and mostly assume stationary (*i.e.*, time independent) behavior. Consequently, in this chapter, we use empirical data to identify the main characteristics (*e.g.*, self-similarity, nonstationarity) of the communication workload of real CPS. Starting from the complex characteristics of CPS workloads, we present a novel statistical physics inspired model which is used to define a new optimal control problem that not only accounts for the observed self-similarity and nonstationarity properties of the CPS workload, but also allows for accurate predictions on CPS dynamical trajectories during the optimization process. This opens new venues for CPS design and optimization for real life applications. As a concrete example, we discuss the application of this theory to the control algorithm of a pacemaker.

7.1 State of The Art in CPS Design and Optimization

There is an increasing concern for bringing computation and communication together in order to design efficient CPS [112][130][149]. These systems consist of networks of embedded computation and communication devices, as well as sensors, which are used to monitor and measure various physical processes taking place on electrical power grids, transportation and traffic roads, communication and financial networks, medical devices, smart buildings. Hence, CPS need to be dependable, safe, reliable, efficient, real-time, yet secure [93][130][147].

We expect that future CPS will help us define new communication and interaction protocols that will provide better control over physical processes. For instance, several research efforts focusing on

how cell phone technology can be used to harness social tracking and knowledge fusion are now well under way [90][108][109]. The principles of community sensing through privately held sensors (*e.g.*, GPS devices, embedded cell phones) are eloquently described in [90], together with several privacy issues. Such community sensing networks can be adequately used for road traffic monitoring. A more concrete incarnation of CPS for traffic optimization is presented in [109], which describes how smart-phones could be employed to sense the environment and transmit information about it to various traffic decision centers. Besides sensing infrastructure, a congestion control protocol has been proposed in [4] which reduces communication traffic based on the importance of collected data and desired estimation error.

From a different perspective, it is crucial to design highly reliable and powerful defense systems that are fault-tolerant to both natural disasters and terrorist attacks. The analysis of CPS features can optimize the design of power grids and oil/gas transportation pipes [130]. Last but not least, CPS have the goal not only to help us design and build environmentally friendly and energy efficient (smart) buildings [86], but also to cater for the changing needs of contemporary society at a time when we are increasingly concerned about dwindling natural resources.

Building such CPS requires a new science of characterizing and controlling dynamic processes across heterogeneous networks of sensors and computational devices. This science needs to bridge the gap between real-time computing and signal processing techniques with distributed and/or self-organizing control of heterogeneous wireless sensor networks and embedded systems. Nevertheless, such a new science cannot rely solely on reductionism and linear control paradigms that represent the norm nowadays.

Figure 7.1 summarizes these challenges and puts forth a statistical physics vision to CPS design. To solve many of the challenges we face today, CPS have to sense and measure various physical processes like heart rate, weather changes, *etc.* (see the bottom of the pyramid). These measurements are further digitized and communicated to various decision centers as varying workloads which represent the Achille's heel for CPS design (middle part of pyramid in Figure 7.1). This is because the accurate characterization of workload properties (*e.g.*, self-similarity, nonstationarity) remains a big challenge for CPS optimization and end users satisfaction (top of pyramid).

Given all these challenges, we argue that several recurrent CPS design problems can be solved in an elegant manner if we resort to statistical physics concepts. Indeed, as shown in Figure 7.1, statistical physics approaches can help by allowing us to describe the CPS workload and its intermediate states (*e.g.*, queue occupancy, node-to-node latency) via fractal-type master equations. Consequently, taking into consideration the space and time features of the CPS workloads via such master equations allows us to formulate various optimization problems such as resource allocation, task mapping and scheduling (top of the pyramid in Figure 7.1). By carefully allocating the buffers size and the bandwidth between sensors and data centers, we can also accelerate the decision-making process and prevent data loss or delay which could otherwise cause life-threatening situations or increase economical costs. These issues are discussed next.

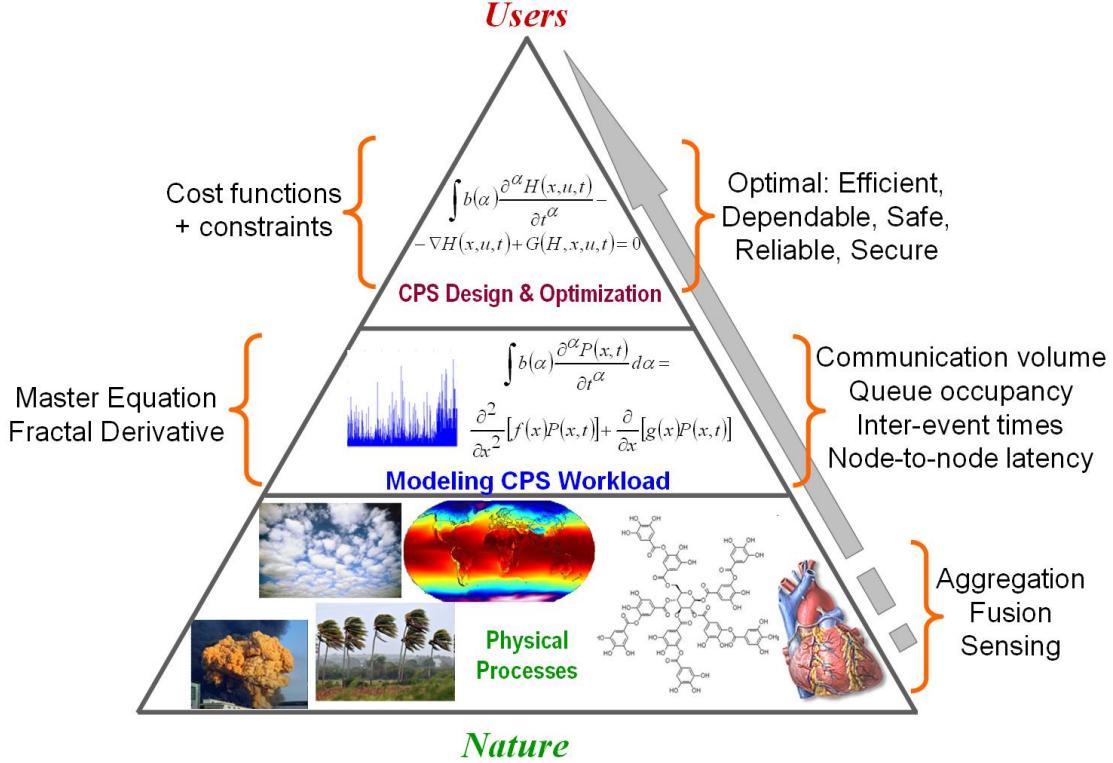


Figure 7.1: CPS pyramid: The pyramid foundation summarizes the large set of physical processes of interest to the CPS community. Insights about physical processes gained through sensing, information fusion and aggregation lead to complex heterogeneous CPS workload. Accurate modeling of workloads enables the design of optimal CPS architectures that may improve our life. Similarly, to statistical physics which has been successful in explaining nature, the CPS workloads are modeled via master equations which later are used in defining various optimization problems of interest to the end users.

7.2 Incorporating Time as Essential Component in Future Computing Platforms

For decades, the science of systems design has tacitly assumed that workloads can be modeled via linear time invariant equations. We argue that this situation has to change and the major developments in statistical physics (*e.g.*, master equation, path integrals, renormalization group theory) developed specifically for processes characterized by strong fluctuations, pseudo-periodicity, and long range memory, should become essential tools for designing future CPS. More precisely, we propose a new formalism that allows not only to estimate the correlation structure observed in CPS traffic traces (which is mainly the core of statistics and estimation theory), but also incorporate such characteristics into some dynamical state equations describing the overall system behavior.

In order to discuss the mathematical underpinnings of CPS workload modeling, let us first define some important parameters. As shown in Figure 1.2.a, various types of sensors monitor diverse physical processes (*e.g.*, volcanic activity, heart rate, car density, CCN concentration), then

aggregate/fuse the measured data into packets and communicate this information as a workload to specialized data centers for further analysis. For instance, a bio-implantable system-on-chip can monitor the heart rate by constructing a time series based on its electrical activity. Using this physiological information, the bio-implantable device can analyze the behavior exhibited by the fluctuations in the R-R intervals and decide to provide more or less electrical stimuli to the heart. Similarly, airplanes in flight can sense the movement of clouds or collect pollution measures (*e.g.*, CO₂, CCN concentration) and communicate it to data centers for weather prediction and climate change analysis.

Obviously, CPS data get sent over the shared communication infrastructure (*e.g.*, wireless networks, internet), so the CPS workloads typically consist of a mix of many types of data. Let us denote by variable $x(t)$ the stochastic process characterizing the CPS workloads (*e.g.*, communication volume, packet delays, *etc.*). Starting from the CPS characteristics (*e.g.*, self-similarity, nonstationarity), the master equation¹ [80][166] describing the dynamics of the stochastic process $x(t)$ can be written as follows:

$$P(x, t|\alpha) = P(x_0, t_0) + \int_{t_0}^t \int_0^\infty w(x-y, t-\tau) P(y, \tau|\alpha) dy d\tau \quad (7.1)$$

where $P(x, t|\alpha)$ denotes the conditional probability of finding the system at time t in a particular state x (*e.g.*, x can represent the level of buffer occupancy, the communication volume between two nodes, the time it takes until a computation request is completed, *etc.*) for a given fractal dimension α , $P(x_0, t_0)$ is the initial condition, and $w(x-y, t-\tau)$ is the kernel probability capturing the dependency of the evolution of probability $P(x, t|\alpha)$ on the memory of the stochastic process $x(t)$.

For instance, the atmospheric measurements made during flights can be aggregated with road traffic information from cars into heterogeneous workloads and then sent via satellite or intermediate nodes to data centers for further processing. In this case, the stochastic process $x(t)$ denotes the amount of information communicated at a specific time. The aggregation process which implies changes in both magnitude and timing patterns of $x(t)$ is captured via the probability $w(x-y, t-\tau)$. Note that this distribution depends on the *inter-event times* (*i.e.*, the difference between two consecutive time stamps when the process $x(t)$ takes different values). For power law distributed inter-event times, the kernel takes the following form $w(x-y, t-\tau) = \nu(x-y)(t-\tau)^{\alpha-1}/\Gamma(\alpha)$, where $\nu(x-y)$ denotes the state transition probabilities and $\Gamma(\alpha)$ is the Gamma function [128]. Of note, the worst case analysis can be retrieved from this formulation by computing the probabilities of rare events (*e.g.*, obtaining the maximum of a random variable) via high-order moments analysis.

Due to the inherent multi-fractal nature of many physical processes, the CPS stochastic processes $x(t)$ can also exhibit a complex self-similar behavior which can be captured via a distribution

¹The master equation is used in statistical physics to characterize through a dynamical equation the evolution of the probability distribution $P(x, t)$ over a set of states.

function $b(\alpha)$ ² of scaling exponents. By multiplying with $b(\alpha)$ both sides of Eq. 7.1 and integrating over the set of fractal dimensions, we can get the following relation for the probability $P(x, t)$ of reaching state x at time t :

$$P(x, t) = P(x_0, t_0) + \int_{\alpha_{min}}^{\alpha_{max}} \int_{t_0}^t \frac{(t-\tau)^{\alpha-1}}{\Gamma(\alpha)} \int_0^\infty \nu(x-y) P(y, \tau|\alpha) dy d\tau \quad (7.2)$$

Using the definition of fractional order operator of integration and derivation [82][118][136], we can show that:

$$\int_{\alpha_{min}}^{\alpha_{max}} b(\alpha) \frac{\partial P(x, t)}{\partial t^\alpha} = \int_0^\infty \nu(x-y) P(y, t) dy \quad (7.3)$$

Next, we assume that the transition probabilities $\nu(x-y)$ can be described as shown in Figure 7.2 and, thus, the right hand side of Eq. 7.3 takes the following form:

$$\begin{aligned} \int_0^\infty \nu(x-y) P(y, t) dy &= a_1 f_1(x-1) P(x-1, t) + a_2 f_2(x+1) P(x+1, t) - \\ &\quad - (a_1 f_1(x) + a_2 f_2(x)) P(x, t) + \int_0^\infty \frac{\gamma(x-y)}{h(y, t)} P(y, t) dy \end{aligned} \quad (7.4)$$

where $a_1 f_1(x-1)$ and $a_2 f_2(x+1)$ are the transition probabilities from states $(x-1)$ and $(x+1)$ to state x (the bottom transitions in Figure 7.2), $a_1 f_1(x) + a_2 f_2(x)$ represents the probability of remaining in the same state x , and $\gamma(x-y)/h(y, t)$ denotes the memory kernel reflecting the intrinsic multiplicative noise that may affect the evaluation of transition probabilities [53]. In a hypothetical situation when the stochastic evolution of the stochastic process $x(t)$ is perfectly characterized, the multiplicative noise (*i.e.*, $\gamma(x-y)/h(y, t)$) becomes zero.

According to Eq. 7.3 and Eq. 7.4, and using the finite difference method [128], the generalized master equation 7.1 characterizing the evolution of the CPS workload $x(t)$ can be expressed as follows:

$$\begin{aligned} \int_{\alpha_{min}}^{\alpha_{max}} b(\alpha) \frac{\partial P(x, t)}{\partial t^\alpha} &= \frac{\partial^2}{\partial t^2} [(a_1 f_1(x) + a_2 f_2(x)) P(x, t)] - \\ &\quad - \frac{\partial}{\partial t} [(a_1 f_1(x) - a_2 f_2(x)) P(x, t)] + \int_0^\infty \frac{\gamma(x-y)}{h(y, t)} P(y, t) dy \end{aligned} \quad (7.5)$$

To capture the multi-fractal features of the stochastic process $x(t)$, the first term in Eq. 7.5

²The multifractal spectrum $b(\alpha)$ can be estimated either as a histogram of fractal dimensions where each fractal dimension is obtained via box-counting algorithm [155] or by the method of moment estimation via the Legendre formula [102].

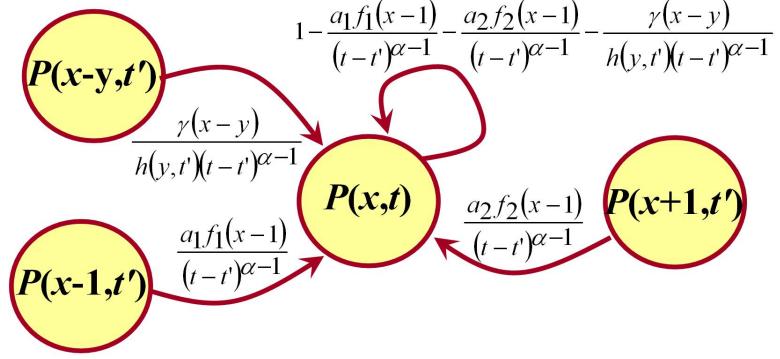


Figure 7.2: Graphical representation of transition probabilities assumed to govern the evolution of the stochastic process $x(t)$. In the absence of uncertainty, the transition probability from state $(x - y, t')$ can be neglected and the noise coefficient $\gamma(x - y) = 0$.

represents the dynamics of CPS workload as a weighted sum of fractional derivatives. Note that a fractional derivative implies that the underlying inter-event times $(t - \tau)$ associated with the stochastic process $x(t)$ are power law rather than being exponentially distributed. For exponential inter-event times, the first term reduces to the classical first order time derivative of $P(x, t)$ with respect to time. The second and third term capture the magnitude of fluctuations in the stochastic process $x(t)$. Finally, the last term models the multiplicative noise that comes from system interaction with the environment. One can note that if $b(\alpha) = \delta(\alpha - 1)$ and $\gamma(x) = 0$, Eq. 7.5 reduces to the normal diffusion equation for the evolution of $P(x, t)$ [12].

From a practical standpoint, the existence of multi-fractal behavior requires, as will become clearer in the next section, new control strategies based on non-linear state equations. To better emphasize this aspect, one can obtain from Eq. 7.5 a Langevin-like dynamical equation characterizing the state of stochastic process with multiplicative noise:

$$\int_{\alpha_{min}}^{\alpha_{max}} b(\alpha) \frac{\partial x(t)}{\partial t^\alpha} = f(x, t) + h(x, t)\eta(t) \quad (7.6)$$

where $f(x, t)$ is given by the following relation:

$$f(x, t) = a_2 f_2(x) - a_1 f_1(x) - \int_0^\infty \frac{\partial [(a_1 f_1(x) + a_2 f_2(x)) P(x, t)]}{\partial x} dx \quad (7.7)$$

Note that for the particular case of $b(\alpha) = \delta(\alpha - 1)$ (*i.e.*, no fractal behavior in time), the linear state space dependence $f(x, t) = a(t)x(t)$ and $h(x, t) = 0$ (zero noise source), so Eq. 7.6 reduces to a classical linear control state equation. Equation 7.6 is used in the next section to define the general problem of optimal control for CPS.

7.3 Dynamic Optimization Methodologies for Cyber-Physical Systems

The successful description of CPS workloads via the new formalism we propose should allow for better optimization and control methodologies. From an engineering perspective, the overarching goal of CPS design is to minimize some specific cost function (*e.g.*, power consumption, fuel consumption, *etc.*) subject to various resource limitations, performance, and reliability constraints. As shown in Figure 7.3, due to the intrinsic continuous interaction of CPS with physical environment, the main tasks of computation, communication and control cannot be modeled, analyzed, and optimized in isolation, but instead several uncertainty and noise sources need to be considered as well. Thus, we formulate the new problem of CPS design (which is also presented in Figure 7.3) as a general (stochastic) optimization problem:

$$\begin{aligned} & \min_{u(t)} \int_{x \in X} \int_{t_{initial}}^{t_{final}} c(x(t), u(t), d(t), t) p(x(t)|u(t)) dt dx \\ & \text{such that } u_{min} \leq u(t) \leq u_{max}, \quad k = 1 \div N, \quad x_i(0) = x_i^0, \quad i = 1 \div M \\ & \int_{\alpha_i^{min}}^{\alpha_i^{max}} b(\alpha) \frac{\partial x_i(t)}{\partial t^{\alpha_i}} = f(x_i, x, t) + g(u, t) + h(x_i, t) \eta_i(t) \end{aligned} \quad (7.8)$$

where $c(x(t), u(t), d(t), t)$ represents the cost objective as a function of the states of the system $x_i \in \Xi$, $i = 1 \div M$, the input control signals $u_k \in \Psi$, $k = 1 \div N$, and the reference/ demands signals $d_j \in \Lambda$, $j = 1 \div L$, $p(x(t)|u(t))$ the conditional probability of finding the CPS in state $x(t)$, given the control scenario $u(t)$; u_{min} and u_{max} are the lower and upper bounds on the control signals $u_k \in \Psi$, $k = 1 \div N$ α_i and $b(\alpha_i)$ are the fractional order derivatives and the distribution of fractal dimensions characterizing the dynamics of $x_i \in \Xi$, $i = 1 \div M$, $f(x_i(t), d(t), t)$ is the generic function describing the evolution of system states $x_i \in \Xi$, $i = 1 \div M$, $g(u(t), t)$ represents the dependency between the control signals $u_k \in \Psi$, $k = 1 \div N$ and the stochastic process $x_i \in \Xi$, $i = 1 \div M$ of interest, $h(x_i(t), t)$ is the multiplicative noise amplitude that depends on the actual state of the system, and $\eta(t)$ is the noise coming from a fluctuating environment (*e.g.*, white noise).

To give some intuition about the sources of this noise, we can, for instance, imagine that under the impact of some disruptive news, a patient whose heart is controlled via a pacemaker experiences a ventricular arrhythmia which results in less blood pumped through the arteries. These changes in the blood flow can only be modeled as some multiplicative noise [167]. Moreover, this arrhythmia indirectly affects not only the heart rate, but also the workload communicated between the pacemaker and the decision center. Similar cases in which the multiplicative noise can affect the CPS workload directly or indirectly can be found in mobile communication channels due to air turbulence, reflection, refraction or multi-path effects (multiplicative interference) [5][91], in optical channels due to amplitude vector rotation [153].

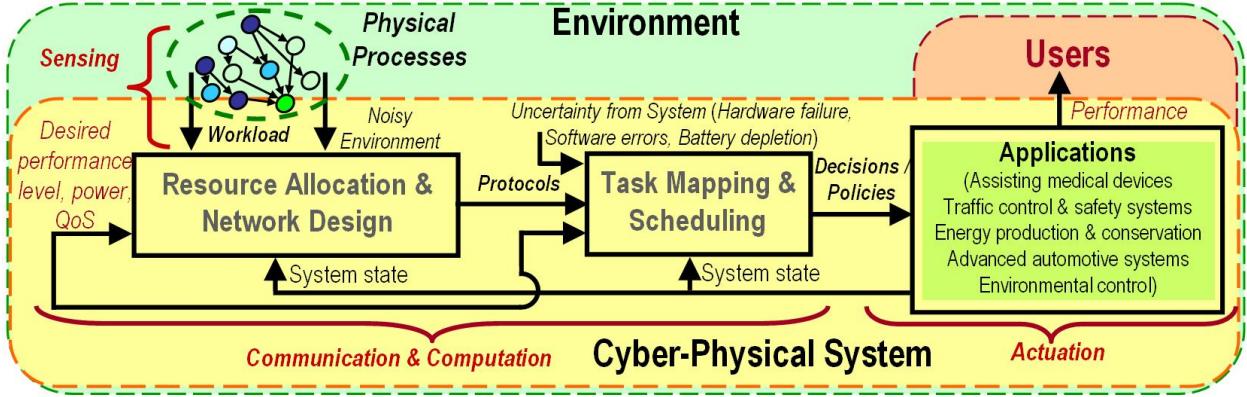


Figure 7.3: Main challenges of CPS design: Resource allocation and network design to accurately sense a set of physical processes within a noisy environment, task mapping and computation scheduling to determine the optimal control strategies that allow satisfying the performance constraints.

7.4 A Dynamic Optimization Case Study: Fractal Optimal Control for Bioimplantable Devices

Up to this point, the discussion has focussed on formulating the CPS design and optimization problems at large. Now let us examine how these concepts can be used for better future CPS-based health care systems. Besides engineering requirements (*e.g.*, high degree of adaptiveness, autonomy, efficiency, functionality, reliability, safety, and usability), the health care CPS need to maximize the patients quality-of-life, while minimizing the intrinsic costs related to patient hospitalization. For instance, given that cardiac diseases are the number one killer around the globe, it is necessary to have accurate bioimplantable devices capable of monitoring heart rate, communicate with medical experts/devices and actuate in real time when heart is misbehaving.

In many respects, there is an urgent need for health care CPS. On the same time, the health care CPS development is stalled due to the lack of a coherent theory that comprehends, conjoins, and coordinates the cyber and physical resources in a unique, efficient and robust approach. One such CPS challenge example is represented by the artificial pacemaker³ consisting of both analog components (*i.e.*, sense amplifiers to monitor information about the heart rate activity and pacing output circuitry) and digital components (*i.e.*, microprocessors and memory blocks for actuating pacing events) [78].

Starting from these overarching ideas, our contribution is three fold [34]: First, we propose a more accurate modeling of the heart rate dynamics via fractional differential equations. Second, we formulate the rate adaptive pacing problem as a model predictive control approach seeking to solve iteratively a constrained finite horizon optimal control with fractal state equations. Third, we compare and contrast various control approaches for pacemakers and give a sense of the hardware implementation complexity of such a fractal controller. Although, in this work, we constructed a

³An artificial pacemaker senses the heartbeats and triggers electrical impulses to the heart in order to regulate the heart rhythm.

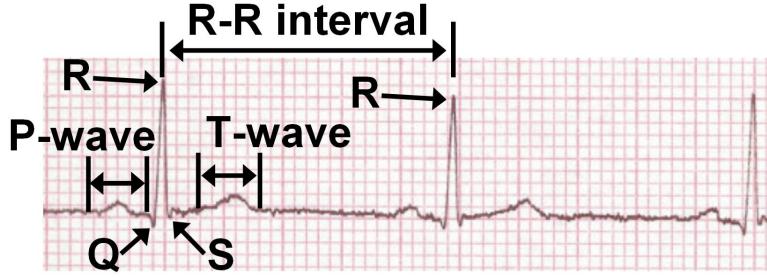


Figure 7.4: A short electrocardiogram showing the heart activity of three beats and the *P*-wave (atrial depolarization), *QRS* complex (ventricles depolarization) and *T*-wave (ventricles repolarization). The $R - R$ interval is the time interval between two consecutive heart beats represented in the electrocardiogram by two consecutive *R*-waves. The variability observed in the $R - R$ intervals is essential for medical diagnosis.

fractal state space model representation for the $R - R$ intervals, the formalism can be used for other physiological processes as well.

To better illustrate the importance of controlling the $R - R$ intervals, Figure 7.4 shows a short electrocardiogram of the heart activity. The $R - R$ interval is defined as the interval of time between two consecutive heart beats and it is a significant metric for deciding whether a person is suffering from a cardiac disease.

The importance of fractal behavior observed in heart rate variability has been ignored by state-of-the-art pacing algorithms. In this section, we build a new theory based on the experimental observations that heart rate processes display a fractal behavior and model it via fractional differential equation. This represents a major departure from the traditional modeling approaches used in control (dynamic) optimization field. The newly proposed fractional calculus description of heart rate processes is encapsulated into an optimal control framework which seeks to determine a certain pacing frequency such that the $R - R$ intervals remain within a predefined set of reference values. In other words, our approach tries to bring the magnitude of $R - R$ intervals to a given reference value from either very large or very small values which are signs of heart (*i.e.*, bradycardia or tachycardia) diseases. For exemplification purposes, we consider the optimal control problem with two different cost functions; however, the proposed formalism can be easily extended to other cost functions and control signals (*e.g.*, magnitude of the pacing voltage applied to either atria or ventricles).

Next, we introduce a finite horizon fractal optimal control approach to heart rate regulation problem. More precisely, given an initial time t_i and a final time t_f , the goal of the optimal controller is to find the optimal control signal, *i.e.*, the frequency of the pacing events, which minimizes the quadratic cost of observing deviations in either the magnitude of $R - R$ intervals or heart rate activity from a predefined reference value, as well as the magnitude of pacing frequency, as shown

below [34]:

$$\min_f \int_{t_i}^{t_f} \left\{ \frac{w}{2} \left(y(t) - y^{ref}(t) \right)^2 + \frac{z}{2} f^2(t) \right\} dt \quad (7.9)$$

subject to the following constraints:

$$\frac{d^\alpha y(t)}{dt^\alpha} = a(t)y(t) + b(t)f(t) \quad (7.10)$$

$$0 < y^{min} \leq y(t) \leq y^{max} < 1 \quad (7.11)$$

$$y(t_i) = y_0 \quad y(t_f) = y_1 \quad (7.12)$$

$$f^{min} \leq f(t) \leq f^{max} \quad (7.13)$$

where $y(t)$ represents the heart rate activity seen as a state variable, $y^{ref}(t)$ denotes the reference values that need to be achieved in terms of heart rate activity, $f(t)$ is the pacing frequency, w and z are the weighting coefficients for the quadratic error and magnitude of the control signal, respectively, in the cost function, α is the exponent of the fractional order derivative characterizing the dynamics of the heart rate activity $y(t)$, $a(t)$ and $b(t)$ are weighting coefficients for the heart activity and pacing frequency, y^{min} and y^{max} are the minimum and maximum bounds on heart rate activity $y(t)$, $y(t_i)$ is the initial condition, $y(t_f)$ is the final condition, f^{min} and f^{max} are the minimum and maximum allowed bounds on pacing frequency $f(t)$.

By focussing on the squared difference between the actual and the reference value in Eq. 7.9, the optimal controller tries to minimize the chances of either positive or negative deviations from the predefined reference. In other words, the cost functional in Eq. 7.9 penalizes for any deviations from the reference value and large magnitudes in the control signal.

The use of the integral of squared difference between the actual and the reference heart rate is also attractive because of two reasons: First, it simplifies to linear equations when evaluating the optimality conditions. Second, the integral of squared error is in general robust to parameter variations. Note that unlike other general formulations of optimal control, in this setup we have to deal with very specific initial and final values summarized in Eq. 7.11. Consequently, the role of the controller is to determine the right pacing frequency which drives the system from one initial state (labelled as life-threatening) to a final state (labelled as safe).

Note also that in both Eq. 7.11 and Eq. 7.13, we impose minimum and maximum bounds on both the accepted $R - R$ intervals and the delivered *pacing frequencies*. These bounds are necessary to be imposed because they prevent the optimal control algorithm from driving the heart muscle system at excessive pacing rates.

To solve the above optimal control problem, we use concepts from the optimization theory and

construct first the Lagrangian functional, $L(y, f, \lambda, \beta_1, \xi_1, \beta_2, \xi_2)$, as follows:

$$L(y, f, \lambda, \beta_1, \xi_1, \beta_2, \xi_2) = \int_{t_i}^{t_f} \left\{ \frac{w}{2} \left(y(t) - y^{ref}(t) \right)^2 + \frac{z}{2} f^2(t) + \beta_1 (f - f^{min} - \xi_1) \right\} dt + \\ + \int_{t_i}^{t_f} \left\{ \beta_2 (f^{max} - f - \xi_2) - \lambda \left[\frac{d^\alpha y(t)}{dt^\alpha} - a(t)y(t) - b(t)f(t) \right] \right\} dt \quad (7.14)$$

where λ , β_1 , and β_2 are the Lagrange multipliers associated with the dynamical state equation for $y(t)$ and the constraints imposed on the control variable f , ξ_1 and ξ_2 are the slack variables needed to transform the inequality bounds into equality constraints on the control variable f .

By expanding the Lagrange function in Eq. 7.14 via the Taylor formula and considering that it attains its minimum in the vicinity of $\tau = 0$, *i.e.*, $\partial L/\partial \tau = 0$, we obtain the following relations:

$$\frac{\partial L}{\partial y} + {}_t D_{t_f}^\alpha \frac{\partial L}{\partial {}_t D_t^\alpha y} = 0, \quad \frac{\partial L}{\partial f} = 0, \quad \frac{\partial L}{\partial \lambda} = 0, \quad \frac{\partial L}{\partial \beta_1} = 0, \quad \frac{\partial L}{\partial \beta_2} = 0 \quad (7.15)$$

where ${}_t D_t^\alpha$ and ${}_t D_{t_f}^\alpha$ represent the fractional derivatives operating backward and forward in time, respectively.

In order to solve the relations in Eq. 7.15, we discretize the interval $[t_i, t_f]$ into N intervals of size $(t_f - t_i)/\Delta t$ and use the formula in Eq. 1.4 to construct a linear system as follows:

$$\sum_{i=0}^k \frac{(-1)^i}{\Delta t^\alpha} \binom{\alpha}{i} y((k-i)\Delta t) - a(k\Delta t)y(k\Delta t) + \frac{b(k\Delta t)^2}{z(k\Delta t)} \lambda(k\Delta t) + \\ \frac{b(k\Delta t)\beta_1(k\Delta t)}{z(k\Delta t)} - \frac{b(k\Delta t)\beta_2(k\Delta t)}{z(k\Delta t)} = 0 \quad k = 1 \div N \quad (7.16)$$

$$\sum_{i=0}^{N-k} \frac{(-1)^i}{\Delta t^\alpha} \binom{\alpha}{i} \lambda((k+i)\Delta t) - w(k\Delta t) [y(k\Delta t) - y^{ref}(k\Delta t)] - a(k\Delta t)\lambda(k\Delta t) + \\ + \frac{\lambda(N\Delta t)(t_f - t_i - k\Delta t)^{-\alpha}}{\Gamma(1-\alpha)} = 0 \quad k = N-1 \div 0 \quad (7.17)$$

$$\frac{\beta_2(k\Delta t) - \beta_1(k\Delta t)}{z(k\Delta t)} - \frac{b(k\Delta t)}{z(k\Delta t)} \lambda(k\Delta t) - \xi_1(k\Delta t) = f^{min}, \quad k = 1 \div N \quad (7.18)$$

$$\frac{\beta_2(k\Delta t) - \beta_1(k\Delta t)}{z(k\Delta t)} - \frac{b(k\Delta t)}{z(k\Delta t)} \lambda(k\Delta t) + \xi_2(k\Delta t) = f^{max}, \quad k = 1 \div N \quad (7.19)$$

In summary, the constrained finite horizon fractal optimal control defined in Eq. 7.9 through Eq. 7.13 reduces to solving a linear system in Eq. 7.16, Eq. 7.17, Eq. 7.18, and Eq. 7.19.

An important aspect of optimal control approaches is to rely on accurate models of the dynamical system or state variables of interest to the designer. Consequently, in what follows, we first estimate the parameters corresponding to a non-fractal and a fractal model and analyze the goodness-of-fit of each approach. More precisely, we perform a hypothesis testing problem by investigating whether the observed data can be modeled via a specific model. In this context, the

Heart rate	Classical (non-fractal) model				Fractal model				
	time series ID	a	b	P-value	Test statistic	α	a	b	P-value
1	0.9665	0.0267	0	61.127	0.6226	0.1613	-0.1284	1	0.2754
2	0.979	0.0193	0	30.031	0.7762	0.0355	-0.0326	1	0.2564
3	0.9513	0.0403	0	120.880	0.707	0.1255	-0.1038	0.51	0.3182
4	0.9525	0.0409	0	73.945	0.7258	0.1105	-0.0951	1	0.2544
5	0.9663	0.0286	0	88.339	0.699	0.0643	-0.0546	1	0.2714

Table 7.1: Comparison between a non-fractal model (1-step ARMA) and a fractal model (see Eq. 7.10 in terms of the estimated parameters and the goodness-of-fit obtained for five time series of heart rate activity. The R-R interval time series are obtained for healthy individuals [126]. Except the time series with ID 3, all other hear rate activity series can be modeled through a fractional order differential equation of the type presented in Eq. 7.10. This is justified by the estimated p-values and test statistics.

goodness-of-fit measures (via the P-value⁴) the discrepancy between the real measurements and the model predictions. Next, we provide a complete numerical analysis of the proposed fractal optimal control problem.

To make the discussion more concrete, we consider two types of models: First, we model the heart rate through a first order differential equation, estimate the corresponding parameters (*i.e.*, a and b) and compute the goodness-of-fit between the actual measurements and the obtained model. Second, we assume that over a finite time interval the heart rate can be modeled through a fractional order differential equation of the type presented in Eq. 7.10, estimate the parameters α , a , and b and report its goodness-of-fit.

Table 7.1 summarizes the estimated parameters and the goodness-of-fit results for both the non-fractal and fractal models proposed to model the heart rate dynamics of healthy individuals. To discriminate in terms of accuracy between the two models, we use the goodness-of-fit described in [18] at 0.05 statistical significance level. This implies that we perform a null hypothesis testing against each model and if the P -value computed for the considered model is below 0.05 level, then with 95% confidence we reject the model as a good fit for the data. Note that this statistical approach proves to be a more robust way of validating models than relying on mean square method [18]. By comparing the P -values in the fourth and ninth columns, we can draw the following conclusions:

- The modeling approach of heat rate processes via a fractional differential equation (as shown in Eq. 7.10) cannot be rejected. The model is superior to the one based on first order derivative for all five heart rate times in both P -values and test statistics.

⁴The P -value is a goodness-of-fit metric. A small P -value (below the significance value) allows us to reject the null hypothesis (*e.g.*, the data follows a certain distribution or can be modeled via an ARMA type with specific parameters).

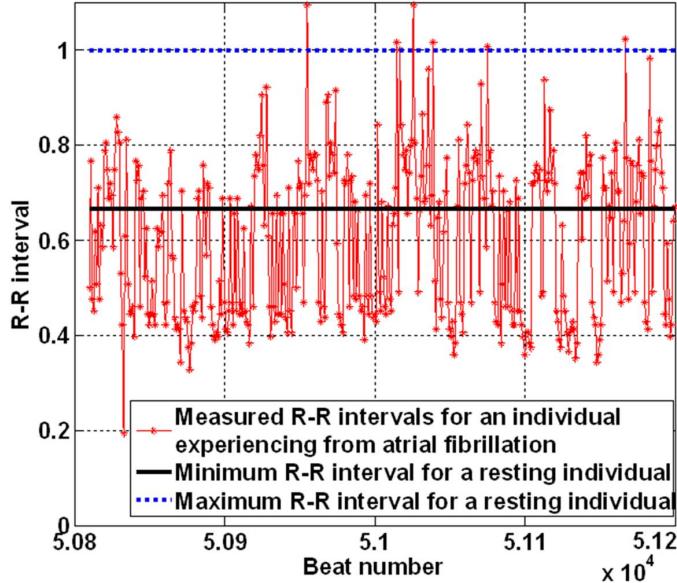


Figure 7.5: Comparison between the measured $R - R$ intervals of an individual suffering from atrial fibrillation and the minimum and maximum bounds on the R-R intervals for a healthy individual at rest. One can clearly observe that the $R - R$ intervals have the tendency of exhibiting small magnitudes. For instance, the first 100 beats exhibit on average $R - R$ intervals of 0.55 seconds with a standard deviation of 0.14 seconds. This can lead to palpitations and so fainting and dizziness.

- The modeling of heart rate dynamics via a first order derivative type of model is strongly rejected by the observed null P -values and high test statistic results.

To illustrate the performance and efficiency of the proposed optimal controller for regulating the pacing frequency of an artificial pacemaker, we consider the heart rate of an individual suffering from atrial fibrillation (see Figure 7.5). To better emphasize the abnormal behavior in the length of $R - R$ intervals, we also plot the minimum (black solid line, corresponding to 0.667 seconds) and maximum (blue dotted line, corresponding to 1 second) bounds for a normal heart rhythm. In addition, we assume that the measured heart rate comes from an CPS infrastructure where the normalized pacing frequency was set to 1 for a fixed interval of time of 56 seconds corresponding to 100 recorded beats. Both the natural and artificial pacing led to an elevated average heart rate of 108 beats per minute. The elevated heart rate is frequently experienced as heart palpitations and can cause fainting and dizziness leading to major injuries. Thus, the role of an adaptive CPS pacemaker is to regulate the pacing frequency in conjunction with the natural pacing coming from the brain in order to keep the heart rate between 60 and 90 beats per minute.

The first step in the analysis is to check which of the two modeling approaches (*i.e.*, the nonfractal one represented by a first order differential equation and the fractal one given by a single fractional order differential equation) is more suitable to be used for capturing the heart rate characteristics exhibited for over 390 heart beats or an interval of time of 235.36 seconds. By relying on the goodness-of-fit algorithm presented in [18], the P -value and test statistics for the integer first order differential equation based model are 0.0018 and 0.3845, respectively. Since we

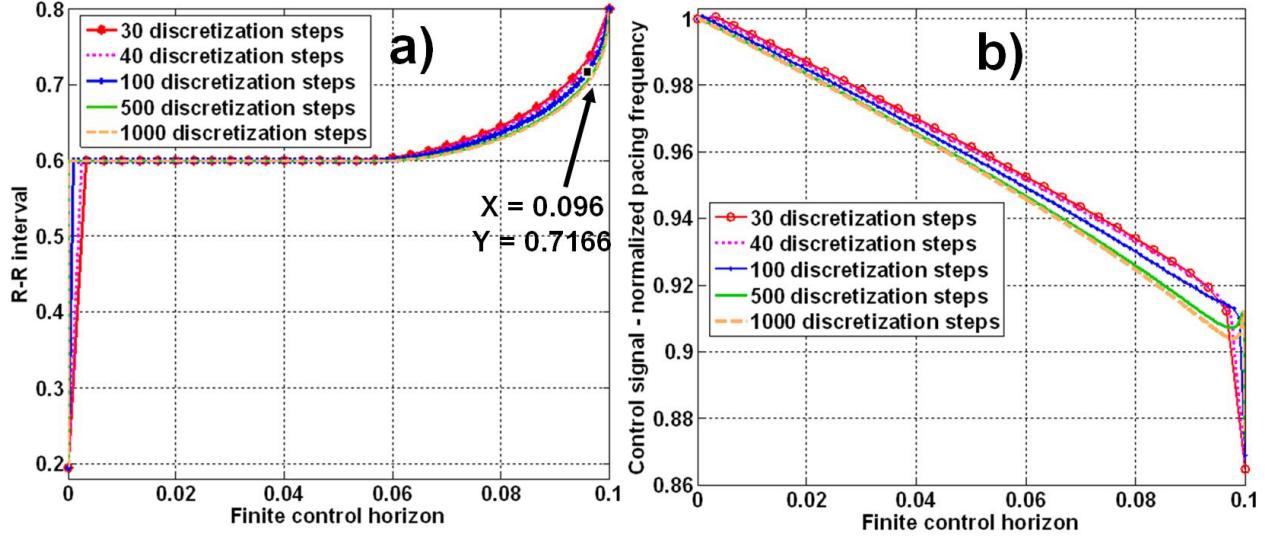


Figure 7.6: Applying the fractal optimal controller, the $R - R$ interval is increased from 0.19 to 0.80 (corresponding to a healthy heart rate of 75 beats per minute) in a finite control horizon of 0.1 seconds. b) Control signal - pacing frequency - necessary to be following by the optimal controller module of the pacemaker to increase the $R - R$ intervals and reduce the heart rate from approximately 100 to 75 beats per minute

performed the null hypothesis testing at 0.05 significance value, based on small P -value of 0.0018 we can reject the integer order differential equation as a good model. In contrast, by applying the same goodness-of-fit algorithm, the P -value and the test statistic for a fractional single order differential equation type of model are 0.8471 and 0.2949, respectively. This shows that for this interval of time, the heart rate and implicitly the $R - R$ intervals can be better modeled via a fractional order differential equation.

Once the parameter identification and goodness-of-fit analysis is completed (and able to validate or invalidate one type of model), the role of the optimal controller in Eq. 7.15 through Eq. 7.16 is to determine the optimal pacing frequency for which the $R - R$ intervals can be increased from the observed 0.20 seconds to 0.80 seconds corresponding to a normal heart rate of 75 beats per minute. Figure 7.6.a shows the impact of considering various discretization steps (*i.e.*, $N = 30, 40, 100, 500$, and 1000 discretization steps) on the $R - R$ intervals for a finite control horizon of 0.1 second. Note that the controller was constrained to find the control signal such that the $R - R$ intervals are between 0.6 and 1 and the pacing frequency between 0.5 and 1. The w and z coefficients in the performance index function shown in Eq. 7.15 were set to 0.1.

Comparison between a non-fractal model and a fractal model (see Eq. 7.16) in terms of the estimated parameters and the goodness-of-fit obtained for five time series of heart rate activity. The $R - R$ interval time series are obtained for individuals experiencing atrial fibrillation [126]. All the above time series can be modeled through a fractional order differential equation as in Eq. 7.16. This is justified by the estimated p-values and test statistics.

One can clearly see that even for small number of discretization steps, the optimal controller is

able to bring the $R - R$ interval from 0.20 to 0.80 seconds for the predefined control horizon. In addition, Figure 7.6.b shows the control signal (pacing frequency) needed to be followed in parallel with the natural pacing coming from the brain to achieve a 0.8 $R - R$ interval or a heart rate of 75 beats per minute. For completeness, the final frequency as a function of the considered number of discretization steps is as follows: 0.8745 for 1000 steps, 0.8732 for 500 steps, 0.8689 for 100 steps, 0.8658 for 40 steps and 0.8647 for 30 steps. Consequently, the loss in accuracy of computing the normalized pacing frequency from fewer discretization steps is approximately 1.1%.

Chapter 8

Conclusion and Future Research Directions

8.1 Major Contributions of this Dissertation

With the goal of transforming the increasing number of information switching devices into exponentially increasing performance, computing platforms ceased to rely on single highly complex microprocessors and embraced the quest for designing thousand core NoC-based architectures. Nevertheless, this is not easily achievable mainly because of several challenges at both application and architectural level. In this thesis, we set forth a statistical physics inspired framework for modeling, analysis and optimization of NoC communication architectures and proposed several dynamic optimization methodologies to reduce the overall power consumption and thermal effects on multi-core systems. Subsequently, we summarize our main contributions:

- Time-dependent and fractal characteristics are two main characteristics that appear in both application and architecture domains. Consequently, in Chapter 2, we proposed a time-dependent probabilistic description of graphs that appear as a result of running a single or a collection of applications on a computing architecture (*i.e.*, the graph edges not only includes dependencies between different application tasks but also communication transactions between processors and memories). Building on such a mathematical formalism, in Chapter 3, we constructed a statistical physics description of the network traffic dynamics via a master equation. Our novel mathematical framework, not only captures the non-stationarity and fractal characteristics observed in synthetic and real world traffic traces, but also enables accurate dynamic predictions of various performance metrics at both coarse- (*e.g.*, network throughput, average packet latency) and fine-level of granularity (*e.g.*, buffer overflow probability, source-to-destination latency exceedance probability, packet waiting time distributions in network buffers, *etc*). In addition, our approach can be used to guide the design space exploration and solve challenging problems such as network buffer allocation¹. Extensive

¹Such an approach starts from the master equation description of the traffic dynamics through each buffer and

experimental results on both synthetic traffic and benchmark applications demonstrate the accuracy and efficiency of the proposed mathematical framework.

- Shrinking transistor sizes, scaling down supply voltages, and increasing clock frequencies lead to a higher sensitivity of nano-circuits to particle strikes, as well as to an increase in the number of timing violation. In order to overcome the negative impact of these factors on NoC reliability, we draw inspiration from virus and/or rumor spreading theory and design the communication among processors as a probabilistic gossiping process. While theories on robustness and coverage (*e.g.*, number of nodes aware of some piece of information) of such virus like communication protocols are ample and sound, there are very few viable approaches to quantifying concrete network performance metrics (*e.g.*, latency), which can guide NoC design and optimization. We proposed a unified mathematical framework for describing such a fault-tolerant communication protocol in both space and time as a collection of branching and annihilating random walks. At the heart of the analytical model lies a master equation describing the state (*i.e.*, number of received messages) of each node in the network and three transition events: packet duplications corresponding to the birth of new random walkers into the network, packet transmissions between neighboring nodes, and packet corruption representing annihilated random walks. Relying on the master equation and its solution, which represents the probability of finding the network in a certain configuration, we were able to derive mathematical expressions for the mean hitting time necessary for a packet to reach a certain destination node from a predefined source. Unlike traditional computer science approaches, which assume the system is in a stationary state and do not take into consideration the time component, our approach enables the definition of dynamic optimization algorithms for future multi-cores, where time plays a fundamental role which cannot be disregarded.
- Besides performance improvement, the fractal nature of NoC traffic has profound effects on dynamic optimization problems such as dynamic power management in voltage/frequency island (VFI) designs or chip temperature regulation. Hence, we developed a fast and efficient algorithm which seeks to determine the optimal supply voltages and operating frequencies of an VFI-based NoC such that the total power consumption and the peak temperature are minimized while satisfying predefined levels of performance (*e.g.*, source-to-destination latency, throughput). Unlike current dynamic voltage and frequency scaling algorithms which ignore the model fitting or parameter identification process, we proposed a linear time parameter identification algorithm for the fractal modeling approach we introduced. We also demonstrated that significant power savings can be achieved under fractal workloads, which cannot be possible by employing the classical optimal control theory.
- CPS integrates computation, communication, and control with physical processes. Physical

seeks to minimize the buffer overflow probability while the source-to-destination latencies are kept under certain bounds.

processes are predominantly non-stationary and require time dependent models for modeling and understanding their behavior. However, most current computing platforms lack proper models for the time component and mostly assume stationary (*i.e.*, time independent) behavior. Unlike traditional approaches, this research starts from the main characteristics (*e.g.*, self-similarity, nonstationarity) of the communication workload of real CPS and builds a theoretical foundation for CPS design and optimization by combining concepts from statistical physics, optimal control and fractional calculus theory. By accurately modeling CPS dynamics through fractional dynamical equations, we were able to formulate a unique framework for CPS design (*e.g.*, resource allocation under fractal demand). Relying on low complexity mathematical techniques, we have demonstrated that fractal optimal controllers can achieve a more stable response and faster convergence time over classical feedback and/or optimal control approaches. Our current work investigates both analytical and numerical techniques to improve the effectiveness of the fractal optimal controller in the CPS context, particularly in the face of increased degree of uncertainty and global computation, communication, and/or energy constraints.

8.2 Beyond Silicon: Future Research Directions

Modeling and analysis of dynamic transport phenomena is a fundamental issue in many research area ranging from transportation systems [70], to communication [120] and biological networks [137]. Consequently, there are several directions in which our research can prove to be very useful.

On one hand, the proposed traffic model implies new (fractal-based) control strategies [102] for dynamic optimization of NoCs which can overcome the limitations of classical queueing and system theory. More precisely, starting from Eq. 4.2, we can model the evolution of buffer utilization via fractal derivatives [102][115] and determine more accurate estimates about the waiting times of packets in the network buffers. This can be used for determining analytical estimates of the probability distribution function of the source-to-destination latencies and so, it can help to formulate dynamic mapping and scheduling algorithms that can be implemented at OS or network level. Also, relying on the concept of fitness distribution one can define hierarchical optimal (traffic aware) routing algorithms that minimize the source-to-destination latency and buffer utilization. This remains as future work.

As will be discussed in Section 8.3, the performance analysis framework developed for stochastic communication problem can be applied to solve diagnostic and drug delivery problems [143]. More precisely, the drug delivery problem needs to quantify the dynamics of multiple random walkers and their biological interactions. Consequently, the concept of hitting time probability can be used to estimate the success of targeted drug delivery.

Another research direction is represented by developing efficient and lightweight control algorithms (of fractal workloads) for cyber-physical systems. Besides the digital aspect presented in this thesis, it would be worth investigating analog implementation solutions of various optimal

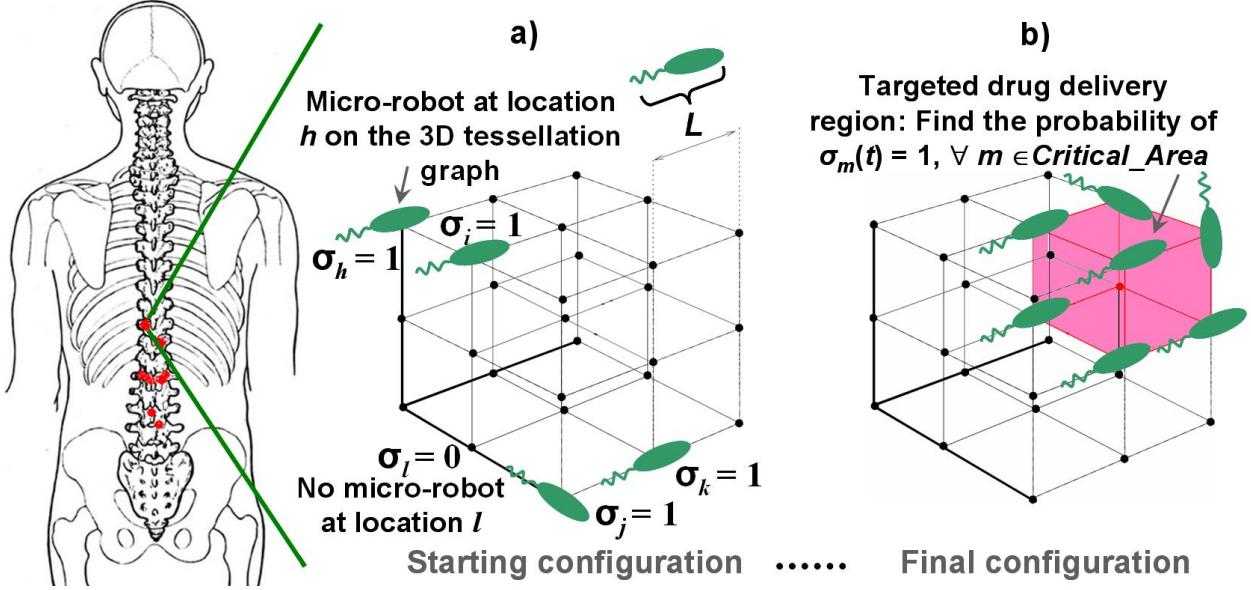


Figure 8.1: Dense network of micro-robotic swarms swim in hard to access regions of the body. The micro-robots dynamics is modeled as a collective behavior of multiple interacting random walks in a 3D graph tessellation of space. Each graph node has associated a binary random variable (σ) denoting whether or not it is occupied by one micro-robot. For detection and health monitoring purposes the goal is to find the time-dependent coverage of the swarm. The goal of the drug delivery problem is to find the probability for the nodes in the disease area (see Figure 8.1.b) to be occupied by micro-robots.

control algorithms such as constrained finite horizon performance tracking, minimum time horizon with performance tracking and constrained control signals or minimum time horizon while power expenditure is maintained within some bounds.

Building on our approach to modeling and analyzing dynamical processes taking place on networks, it is possible to model and optimize various network infrastructures such as roads while taking into account human dynamics [33]. This problem will be briefly discussed in Section 8.4.

8.3 Modeling, Analysis and Optimization of Bacteria-propelled Micro-robotic Swarms with Applications in Medicine

One of the main challenges of modern medicine is the detection of silent progression and migration of various diseases through the human body. For instance, cancer is one of the leading causes of death because in most situations tumors appear and develop undetected by many of the current screening tests or the immune system. Nevertheless, a large body of research in the analysis of tumor angiogenesis suggests that tumors silent progression is most of the time accompanied by an increased demand of nutrients and oxygen [67]. Most of the time these events are not easily detected by the immune system and the cancer cells can corrupt the neighboring and remote organs up to the point where surgery cannot help with a cure.

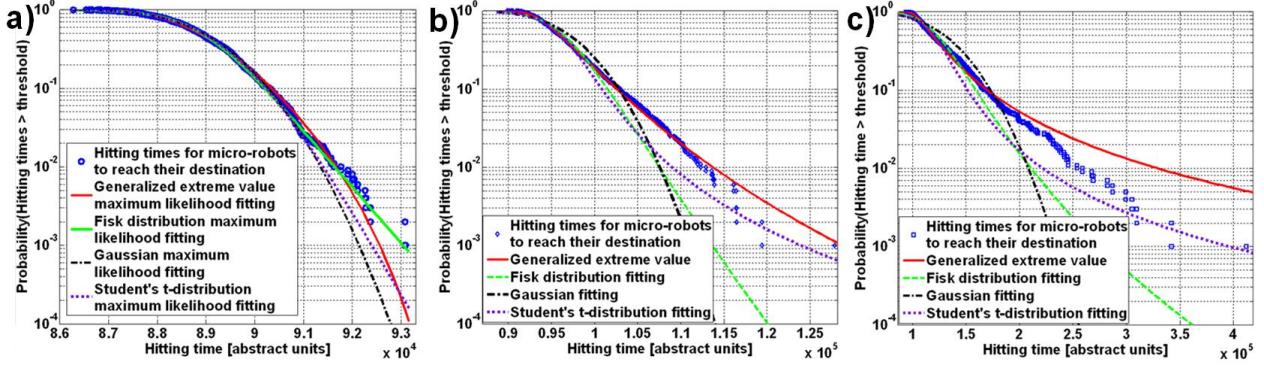


Figure 8.2: Hitting times for all micro-robots to reach their destinations shown for three interaction distances: $5 \mu m$ (5 hops) (a), $10 \mu m$ (10 hops) (b), and $15 \mu m$ (15 hops) (c). Distribution of hitting times is better approximated by means of the generalized extreme value, Fisk or Student's t-distribution rather than by means of Gaussian law when interactions are considered.

Despite these challenges, we can rely on bacteria motility [19] to engineer micro-robotic swarms capable of monitoring, detection and targeted drug delivery within the human body. However, while fabrication of single micro-robots is well underway, the mathematical characterization of the collective dynamics of swarms [62][98][140] of such micro-robots represents a major challenge from the perspective of modeling, analyzing and designing such dense swarms of bacteria for targeted diagnostic and drug delivery.

Building on our work, we plan to propose a mathematical model for capturing the dynamics of a large number (or teams) of self-driven micro-robots (*i.e.*, bacteria propelled capsules) able to swim and access small regions of the human body; this voyage takes place in a non-invasive manner due to micro-robots dimensions [143]. Such engineered micro-robots can perform massively parallel and distributed tasks such as diagnostic or drug delivery. Given the affinity of chemotactic bacteria (*e.g.*, *Serratia Marcescens*) to high oxygen consumption around tumors, the micro-robots can sense and swim through the interstitial spaces towards affected regions. Figure 8.1 shows micro-robotic swarms swimming in the spinal cord and sensing the environment for detecting potential cancer risks.

In order to characterize the dynamics of micro-robotic swarms, we map the movement of multiple micro-robots swarming within a confined 3D region to the problem of tracking the trajectories of multiple simultaneous random walks which can interact at various points in space and time [32]. Towards this end, we first tessellate the 3D space into a graph as shown in Figure 8.1.a. Then, we study the dynamics of multiple random walks contained within a certain region via a master equation which finds the probability $P(\sigma_1, \dots, \sigma_M; t)$ of having multiple random walkers at certain locations on the 3D lattice. As shown in Figure 8.1.a, the binary random variable σ_j indicates whether or not there exists a micro-robot at location j (*i.e.*, $\sigma_j = 1$ if the micro-robot is present and $\sigma_j = 0$ if it's not).

For detection and monitoring purposes, our focus is on finding the coverage and frequency of visiting certain nodes in the 3Ds space by at least one random walker.

In contrast, solving the drug delivery problem requires finding the probability of having a critical number of micro-robots within a specific area (see the highlighted region in Figure 8.1.b).

Note that classical diffusion theory cannot be applied to such a scenario since various hydro-dynamical and chemical interactions are crucially affecting the dynamics of such multiple random walks. This statistical physics approach is meant to capture the collective and competitive behavior of particles and predict the evolution of the swarm as a function of the density of walkers and the strength of their interactions [32]. In short, our mathematical formalism should be able to explain the departure from Gaussian laws towards heavy tail distributions we observe in Figure 8.2 as a function of swarm density and interaction distances. The accurate modeling of trajectories and distances travelled by micro-robotic swarms is of crucial importance for solving the diagnostic and drug delivery problems.

8.4 A Statistical Physics Perspective on Human Dynamics Modeling

Human centric processes refer to a web of interactions that can exist between humans via a networked infrastructure such as internet, roads, airports, *etc.* From communication (letters, telegrams, phone calls, emails, text messages, video chats) to mobility/transportation (travel distances, travel times, train passengers, car traffic density), all such human centric processes have been regarded as being random in nature and so hard to understand, model and optimize.

Nevertheless, with the advent of information technology and availability of huge amount of data about human dynamics, it has been proven that human centric processes (*e.g.*, crowds, car traffic, social networks) exhibit complex, but predictable spatio-temporal patterns. Although beneficial from an optimistic point of view regarding their optimization, the existence of these patterns posses new challenges for both tasks: modeling and optimization of networked infrastructures. For instance, the problem of traffic optimization (*e.g.*, determining the number of directional lanes per road segment, building, opening or closing road arteries as a function of traffic demands) is very complex. On one hand, the complexity is due to fractal characteristics of human behavior (changing preferences for alternative traveling routes, bursts in visiting certain locations). On the other hand, the road structure is not a random set of road segments, but instead a fractal one resulting from an evolutionary optimization (*e.g.*, the necessity to move faster goods between strategic locations).

Given the importance of these complex spatio-temporal patterns of human centric processes, we introduce a new dynamic game meant to capture two essential aspects of both human dynamics and real/virtual networked infrastructure [33].

The first principle is that the cost that is driving the behavior of one agent is not evolving randomly in time, but instead its fractal time is encapsulated via a fractional derivative. The role of the fractional derivative is to weigh each event by the power law of inter-event times. In addition to more accurately model the timing behavior, we also assume that the cost value for each agent can be perturbed by either a multiplicative or an additive noise term. This is motivated by the

fact that some agents are very responsive to changes in the environment while others react more slowly being more circumspect to a change.

The second principle is related to modeling the movement of agents through a fractal structure. To be more concrete, in the case of car traffic problem, we construct a master equation which captures the fractal characteristics of space and time through different fractional derivatives.

These two principle allow us to build a statistical physics self-consistent description of a dynamical game which can be used to model and optimize either the dynamics of agents or the fractal structure itself. The proposed game theoretic formulation not only accounts for the empirically observed scaling laws and patterns in human dynamics, but also opens a new research direction in the field of evolutionary and mean field game theory helping to predict emergent crowd behavior.

8.5 A Non-Equilibrium Approach to Biological Communication and Processing Theory

There is an exponential growing interest in system and synthetic biology not only for understanding the biological processes and organisms, but also for designing communication strategies with cells or group of cells and for performing the so called unconventional computing. Not only that these efforts will transform the face of medicine we know, but to be successful they need a solid theory for representing communication and computation events, or in short for modeling information processing in the biology world.

With the goal of building such a information processing theory (or maybe the biological information theory), we foresee a non-equilibrium approach describing the interactions happening at various levels (*e.g.*, atom, molecule, compound) through master equations. From this point onwards, we can define, similarly to classical information theory, metrics and derive both channel and network capacity information processing bounds. In addition, we can determine which ways are good alternatives to talk to real world biological systems. This envisioned mesoscopic information theory, as I intend to call it, remains a very attractive research direction to pursue. One clear benefit that I foresee is in treating severe diseases such as cancer, age macula degeneration and retinal vein occlusions.

Bibliography

- [1] S. Abe and Y. Okamoto, *Nonextensive Statistical Mechanics and Its Applications*, Springer, 2001.
- [2] O. P. Agrawal, O. Defterli, and D. Baleanu, “Fractional Optimal Control Problems With Several State and Control Variables,” *Journal of Vibration and Control*, pp. 1-10, May, 2010.
- [3] M. Agostinelli, J. Hicks, J. Xu, B. Woolery, K. Mistry, K. Zhang, S. Jacobs, J. Jopling, W. Yang, B. Lee, T. Raz, M. Mehalel, Y. Wang, J. Sanford, D. Pivin, C. Peterson, M. DiBattista, S. Pae, M. Jones, S. Johnson, G. Subramanian, “Erratic Fluctuations of SRAM Cache Vmin at 90nm Process Technology Node,” in *Proceedings of IEEE International Electron Devices Meeting*, pp. 655-658, December, 2005.
- [4] H. Ahmadi, T. Abdelzaher, I. Gupta, “Congestion Control for Spatio-temporal Data in Cyber-physical Systems,” in *Proceedings of International Conference on Cyber-Physical Systems*, Stockholm, Sweden, 2010.
- [5] M. S. Alencar and V. C. da Rocha, *Communication Systems*, Springer, 2005.
- [6] N. Alon, C. K. Avin, M. Kozma, G. Lotker and M. R. Tuttle, “Many Random Walks Are Faster Than One,” in *Proceedings of the 20th Annual Symposium on Parallelism in Algorithms and Architectures (SPAA)*, pp. 119-128, Munich, Germany, June 2008.
- [7] L. Ambrosio, L. Caffarelli, M. G. Crandall, L. C. Evans, N. Fusco, *Calculus of Variations and Non-Linear Partial Differential Equations*, Springer, Mathematics, 2008.
- [8] M. Axtell and M. E. Bise, “Fractional Calculus Applications in Control Systems,” in *Proc. IEEE Nat. Aerosp. Electron. Conf.*, New York, pp. 563-566, 1990.
- [9] J. E. Ayers, *Digital Integrated Circuits: Analysis and Design*, Technology & Engineering, CRC Press, 2004.
- [10] N.T.J. Bailey, *The Mathematical Theory of Infectious Diseases*, Charles Griffin and Company, London, UK, 2nd edition, 1975.

-
- [11] M. Bakhouya, S. Suboh, J. Gaber, T. El-Ghazawi, S. Niar, "Performance Evaluation and Design Tradeoffs of On-Chip Interconnect Architectures," Simulation Modeling Practices and Theory, July 2010.
 - [12] R. Balescu, *Aspects of Anomalous Transport in Plasmas*, Institute of Physics Publishing, 2005.
 - [13] R. Balian and D. ter Haar, *From Microphysics to Macrophysics: Methods and Applications of Statistical Physics*, Springer - Science, 2007.
 - [14] Z. Bar-Yossef and M. Gurevich, "Random Sampling from a Search Engine's Index," in Proceedings of the 15th International Conference of World Wide Web, pp. 367-376, May 2006.
 - [15] A-L. Barabási and H. E. Stanley, *Fractal Concepts in Surface Growth*, Cambridge University Press, Cambridge, 1995.
 - [16] D. Ben-Avraham and S. Havlin, *Diffusion and Reactions in Fractals and Disordered Systems*, Cambridge University Press, 2000.
 - [17] L. Benini, G. De Micheli, "Networks-on-Chips: A New SoC Paradigm," IEEE Computer, vol. 35, January, 2002.
 - [18] J. Beran, *Statistics for Long-Memory Processes*, CRC Press - Mathematics, 1994.
 - [19] H. C. Berg, *Random Walks in Biology*, Princeton Univ. Press, 1993.
 - [20] J. M. Berger and B. B. Mandelbrot, "A New Model for Error Clustering in Telephone Circuits," IBM Journal of Research and Development , vol.7, no.3, pp.224-236, July 1963
 - [21] D. Bertozzi, A. Jalabert, S. Murali, R. Tamhankar, S. Stergiou, L. Benini and G. De Micheli, "NoC Synthesis Flow for Customized Domain Specific Multiprocessor Systems-on-Chip," IEEE Trans. Parallel Distrib. Syst. 16, 2, February, 2005.
 - [22] D. Bertozzi, L. Benini, and G. De Micheli, "Energy-reliability trade-off for NoCs," in Networks on chip, Kluwer Academic, 2003.
 - [23] P. Bogdan and R. Marculescu, "Quantum-like Effects in Network-on-Chip Buffers Behavior," in Proceedings of the 44th annual Design Automation Conference (DAC '07), San Diego, California, U.S.A. June, 2007.
 - [24] P. Bogdan, T. Dumitras, and R. Marculescu, "Stochastic Communication: A New Paradigm for Fault-Tolerant Networks-on-Chip," VLSI Design, vol. 2007, Article ID 95348, 17 pages, 2007. doi:10.1155/2007/95348
 - [25] P. Bogdan and R. Marculescu, "Hitting Time Analysis for Stochastic Communication," in Proceedings of the ACM International Conference on Nano-Networks, Boston, September, 2008.

-
- [26] P. Bogdan and R. Marculescu, “Statistical Physics Approaches for Network-on-Chip Traffic Characterization,” in Proceedings of the 7th IEEE/ACM International Conference on Hardware/software Codesign and System Synthesis (CODES+ISSS ’09), Grenoble, France, October, 2009.
- [27] P. Bogdan, M. Kas, R. Marculescu, and O. Mutlu, “QuaLe: A Quantum-Leap Inspired Model for Non-stationary Analysis of NoC Traffic in Chip Multi-processors,” in Proceedings of the 2010 Fourth ACM/IEEE International Symposium on Networks-on-Chip (NOCS ’10), Grenoble, France, May, 2010.
- [28] P. Bogdan and R. Marculescu, “Workload Characterization and Its Impact on Multicore Platform Design,” in Proceedings of the 8th IEEE/ACM International Conference on Hardware/software Codesign and System Synthesis (CODES+ISSS ’10), Scottsdale, Arizona, U.S.A., October, 2010.
- [29] P. Bogdan and R. Marculescu, “Non-Stationary Traffic Analysis and Its Implications on Multicore Platform Design,” , IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, Special Section on Network on Chip, 30(4), pp. 508-519, April 2011
- [30] P. Bogdan and R. Marculescu, “Towards a Science of Cyber-Physical Systems Design,” in Proceedings of IEEE/ACM Second International Conference on Cyber-Physical Systems (ICCPs), CPS Week, pp.99-108, Chicago, U.S.A., April 2011.
- [31] P. Bogdan and R. Marculescu, “Cyberphysical Systems: Workload Modeling and Design Optimization,” IEEE Design and Test of Computers, vol. 28, no. 4, pp. 78-87, July/Aug. 2011.
- [32] P. Bogdan and R. Marculescu, “Modeling Swarms of Micro-robots for Biological Applications,” 5th q-bio Conference on Cellular Information Processing, Santa Fe, New Mexico, USA, 2011.
- [33] P. Bogdan and Radu Marculescu, “A Fractional Calculus Approach to Modeling Fractal Dynamic Games,” in Proceedings of the 50th IEEE Conference on Decision and Control and European Control Conference (CDC-ECC), Orlando, Florida, USA, December 2011.
- [34] P. Bogdan, S. Jain, K. Goyal and R. Marculescu, “Implantable Pacemakers Control and Optimization via Fractional Calculus Approaches: A Cyber-Physical Systems Perspective,”, in Proceedings of IEEE/ACM Third International Conference on Cyber-Physical Systems (IC-CPS), CPS Week, April 2012.
- [35] P. Bogdan, S. Jain, R. Tornero, and R. Marculescu, Optimal Power Management of Multi-domain Networks-on-Chip under Highly Variable Workloads, in Proceedings of the 6th ACM/IEEE International Symposium on Networks-on-Chip (NOCS’12), Lyngby, Denmark, May 2012.
- [36] E. Bolotin, I. Cidon, R. Ginosar, A. Kolodny, “Cost Considerations in Network-on-Chip,” Integration, the VLSI Journal, Vol. 38, No. 1, pp. 19-42, October, 2004.

-
- [37] D. Bol, R. Ambroise, D. Flandre, and J.-D. Legat, "Interests and Limitations of Technology Scaling for Subthreshold Logic," in IEEE Transactions on Very Large Scale Integration (VLSI) Systems, vol.17, no.10, pp.1508-1519, Oct. 2009.
- [38] S. Borkar, "Thousand Core Chips: A Technology Perspective," in Proceedings of the 44th Annual Design Automation Conference (DAC '07). pp. 746-749, 2007.
- [39] R. Bowles, D. King, J. Held, "Addressing the Challenges of Tera Scale Computing," Intel Technology Journal, vol. 13, Issue 4, December, 2009.
- [40] A. Broder and A. R. Karlin, "Bounds on Cover Time," J. of Theoretical Probability, vol. 2, Issue 1, pp. 101-120, 1989.
- [41] A. Bui and D. Sohier, "On Time Analysis of Random Walk Based Token Circulation Algorithms," in Proceedings of the International School Symposium on Advanced Distributed Systems, Lecture Notes on Computer Science 3563, pp. 63-71, 2005.
- [42] J. L. Cardy and U. C. Tauber, "Theory of Branching and Annihilating Random Walks," Phys. Rev. Lett., vol. 77, no. 23, pp. 4780-4783, 1996.
- [43] A. K. Chandra, P. Raghavan, W.L. Ruzzo and R. Smolensky, "The Electrical Resistance of a Graph Captures its Commute and Cover Times," in Proc. of the 21st annual ACM Symp. on Theory of Computing, pp. 574-586, Seattle, Washington, USA, May 14-17, 1989.
- [44] W. Coffey, Yu. P. Kalmykov and J. T. Waldron, *The Langevin Equation: with Applications to Stochastic Problems in Physics, Chemistry, and Electrical Engineering*, World Scientific - Mathematics , 2004.
- [45] W. Coffey and Yu. P. Kalmykov, *Fractals, Diffusion and Relaxation in Disordered Complex Systems*, John Wiley & Sons - Science, Jun 30, 2006.
- [46] I. Cohen, O. Rottenstreich and I.; Keslassy, "Statistical Approach to Networks-on-Chip," IEEE Transactions on Computers, vol.59, no.6, pp.748-761, June 2010.
- [47] C. Constantinescu, "Trends and Challenges in VLSI Circuit Reliability," IEEE Micro, 2003.
- [48] C. Cooper, A. Frieze and T. Radzik, "Multiple Random Walks in Random Regular Graphs," in SIAM J. Discrete Math., vol. 23, Issue 4, pp. 1738-1761, 2009.
- [49] D. J. Daley and J. Gani, *Epidemics Modelling: An Introduction*, Cambridge University Press, UK, 1999.
- [50] W. J. Dally and C. Seitz, "The Torus Routing Chip," Distributed Processing, vol. 1, pp. 187-196, 1996

-
- [51] W. J. Dally and B. Towles, “Route Packets, Not Wires: On-Chip Interconnection Networks,” in Proceedings of the 38th annual Design Automation Conference (DAC ’01), Las Vegas, Nevada, U.S.A., June, 2001.
- [52] W. J. Dally, “Performance analysis of k-ary n-cube interconnection networks,” IEEE Trans. Comput., vol. 39, no. 6, pp. 775-785, Jun. 1990.
- [53] S. I. Denisov, W. Horsthemke, and P. Hänggi, “Generalized Fokker-Planck equation: Derivation and exact solutions,” European Physical Journal B, vol. 68, Issue 4, pp.567-575, 2009.
- [54] S.N. Dorogovtsev and J.F.F. Mendes, *Evolution of Networks: from Biological Networks to the Internet and WWW*, Oxford University Press, 2003.
- [55] P. G. Doyle and J. L. Snell, *Random walks and electric networks*, Carus Mathematical Monographs, 22, Mathematical Association of America, 1984.
- [56] J. T. Draper and J. Ghosh, “A comprehensive analytical model for wormhole routing in multicomputer systems,” J. Parallel Distrib. Comput., vol. 23, no. 2, pp. 202-214, Nov. 1994.
- [57] T. Dumitras, S. Kerner, R. Marculescu, “Towards on-chip fault-tolerant communication,” in Proceedings of Asia and South Pacific Design Automation Conference, Kitakyushu, Japan, Jan. 2003.
- [58] A. Einstein, *Investigations on the Theory of the Brownian Movement*, Dover Publications Inc., 1956.
- [59] A. Erramilli, M. Roughan, D. Veitch, and W. Willinger, “Self-similar traffic and network dynamics,” Proc. IEEE, vol. 90, no. 5, pp. 800-819, May 2002.
- [60] P. T. Eugster, R. Guerraoui, A. Kermarrec and L. Massoulie, “Epidemic Information Dissemination in Distributed Systems,” Computer, vol.37, no.5, pp. 60- 67, May 2004.
- [61] A. Ganguly, K. Chang, S. Deb, P. P. Pande, B. Belzer and C. Teuscher, “Scalable Hybrid Wireless Network-on-Chip Architectures for Multi-Core Systems,” IEEE Transactions on Computers, vol. 99, 2010.
- [62] V. Gazi and K.M. Passino, *Swarm Stability and Optimization*, Springer Technology & Engineering, 2011.
- [63] C. Gkantsidis, M. Mihail, and A. Saberi, “Random Walks in Peer-to-Peer Networks,” in Proceedings of the 23rd Annual Joint Conference of IEEE Computer Communication Society, vol. 1., pp. 120-130, Mar. 2004.
- [64] W. G. Glöckle and T. F. Nonnenmacher, “A Fractional Calculus Approach to Self-Similar Protein Dynamics,” Biophys. J., vol. 68, no. 1, pp. 46-53, Jan. 1995.

-
- [65] R. Gorenflo, F. Mainardi, and A. Vivoli, “Continuous Time Random Walk and Parameteric Subordination in Fractional Diffusion,” *Chaos, Solitons and Fractals*, vol. 34, pp. 89-103, 2007.
- [66] W. J. Guan, W. K. Tsai, and D. Blough, “An analytical model for wormhole routing in multicomputer interconnection networks,” in Proc. Int. Conf. Parallel Process., pp. 650-654, 1993.
- [67] D. Hanahan and R.A. Weinberg, “The Hallmarks of Cancer,” *Cell*, Vol. 100, Issue 1, pp. 57-70, January 2000.
- [68] J. Held, “From a Few Cores to Many: A Tera-scale Computing Research Overview,” Intel White Paper, 2006. Retrieved from http://download.intel.com/research/platform/terascale/terascale_overviewpaper.pdf
- [69] A. Hemani, A. Jantsch, S. Kumar, A. Postula, and J. Oberg, “Network-on-Chip: An Architecture for Billion Transistor Era,” in Proceedings of the IEEE NorChip Conference, November, 2000.
- [70] R. Herman, T. N. Lam and I. Prigogine, *Kinetic Theory of Vehicular Traffic: Comparison with Data*, Elsevier, 1971.
- [70] R. Hilfer and L. Anton, “Fractional Master Equations and Fractal Time Random Walks,” *Phys. Rev. E*, vol. 51, No. 2, Feb. 1995.
- [72] J. R. M. Hosking, “Fractional Differencing,” *Biometrika*, Vol. 68, No. 1, pp. 165-176, April 1981.
- [73] J. Howard, S. Dighe, Y. Hoskote, S. Vangal, D. Finan, G. Ruhl, D. Jenkins, H. Wilson, N. Borkar, G. Schrom, F. Paillet, S. Jain, T. Jacob, S. Yada, S. Marella, P. Salihundam, V. Erraguntla, M. Konow, M. Riepen, G. Droege, J. Lindemann, M. Gries, T. Apel, K. Henriss, T. Lund-Larsen, S. Steibl, S. Borkar, V. De, R. Van Der Wijngaart, T. Mattson, “A 48-Core IA-32 message-passing processor with DVFS in 45nm CMOS,” in Proceedings of the IEEE International Solid-State Circuits Conference, pp.108-109, 7-11 February, 2010.
- [74] P. Hu and L. Kleinrock, “An analytical model for wormhole routing with finite size input buffers,” in Proc. 15th Int. Teletraffic Congr., Jun. pp. 549-560, 1997.
- [75] J. Hu and R. Marculescu, “Energy- and Performance-Aware Mapping for Regular NoC Architectures,” in IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems, Vol.24, No.4, April 2005.
- [76] Y. Ijiri and H. A. Simon, *Skew Distributions and the Sizes of Business Firms*, North-Holland Pub. Co. - Mathematics, 1977.
- [77] A. Jantsch and H. Tenhunen, *Networks-on-Chip*, Kluwer Press, 2003.

-
- [78] S. A. P. Haddad, R. P. M. Houben and W. A. Serdijn, “The Evolution of Pacemakers,” IEEE Eng. Medicine & Biology, 2006.
- [79] J. Hartmanis and R. E. Stearns, On the Computational Complexity of Algorithms, Transactions of the American Mathematical Society , Vol. 117, pp. 285-306, May, 1965.
- [80] R. Hilfer, *Applications of Fractional Calculus in Physics*, World Scientific, 2000.
- [81] N. G. Van Kampen, *Stochastic Processes in Physics and Chemistry*, Elsevier, 2007.
- [82] A. A. Kilbas, H. M. Srivastava, and J. J. Trujillo, *Theory and Applications of Fractional Differential Equations*, Amsterdam, Netherlands, Elsevier, February 2006.
- [83] J. Klafter, A. Blumen and M.F. Shlesinger, “Stochastic Pathway to Anomalous Diffusion,” Physical Review A, vol. 35, no. 7, pp. 3081-3085, April 1, 1987.
- [84] R. Klages, G. Radons, and I. M. Sokolov, *Anomalous Transport: Foundations and Applications*, Wiley-VCH Science, 2008.
- [85] R. Klages, *Microscopic Chaos, Fractals and Transport in Nonequilibrium Statistical Mechanics*, World Scientific - Science, 2007.
- [86] J. Kleissl and Y. Agarwal, “Cyber-Physical Energy Systems: Focus on Smart Buildings,” in Proceedings of the 47th ACM Design Automation Conference (DAC), Anaheim, 2010.
- [87] H. Kobayashi and B. L. Mark, *System Modeling and Analysis: Foundations of System Performance and Evaluation*, Pearson, 2009.
- [88] A. N. Kolmogorov, A. N. Shiriaev, V. M. Tikhomirov, *Selected Works of A.N. Kolmogorov: Probability Theory and Mathematical Statistics*, Springer, 1992.
- [89] P. L. Krapivsky, S. Redner and E. Ben-Naim, *A Kinetic View of Statistical Physics*, Cambridge University Press - Science, 2010.
- [90] A. Krause, E. Horvitz, A. Kansal and F. Zhao, “Toward Community Sensing,” IEEE Computer Society International Conference on Information Processing in Sensor Networks, 2008.
- [91] E. A. Lee and D. G. Messerschmitt, *Digital Communication*, Kluwer Academic Publishers, 1995.
- [92] H. G. Lee, N. Chang, U. Y. Ogras and R. Marculescu, “On-chip communication architecture exploration: A quantitative evaluation of point-to-point, bus, and network-on-chip approaches,” in ACM Trans. on Design Automation of Electronic Systems, Vol.12, No.3, Aug. 2007.
- [93] E. A. Lee, “CPS Foundations,” in Proceedings of the 47th ACM Design Automation Conference (DAC ’10), 2010.

-
- [94] E. A. Lee and S. A. Seshia, Introduction to Embedded Systems - A Cyber-Physical Systems Approach, <http://LeeSeshia.org>, 2011.
- [95] W. E. Leland and D. V. Wilson, "High time-resolution measurement and analysis of LAN traffic," IEEE Comp. & Comm. Soc., vol.3, 1991.
- [96] J.-B. Lesort, E. Bourrel and V. Henn, "Various Scales for Traffic Flow Representation: Some Reflections," Traffic and Granular Flow, 2003.
- [97] A. Lungu, P. Bose, D. J. Sorin, S. German, and G. Janssen, "Multicore Power Management: Ensuring Robustness via Early-Stage Formal Verification," in Proceedings of the 7th IEEE/ACM intl. conf. on Formal Methods and Models for Codesign (MEMOCODE'09), 2009.
- [98] R. Mahnke, J. Kaupuzs, and I. Lubashevsky, *Physics of Stochastic Processes: How Randomness Acts in Time*, Wiley, 2009.
- [99] F. Mainardi, *Fractional Calculus and Waves in Linear Viscoelasticity*, Imperial College Press, 2010.
- [100] Y. Malevergne and D. Sornette, *Extreme financial risks: from dependence to risk management*, Springer-Verlag, 2006.
- [101] B. B. Mandelbrot and J. W. Van Ness, "Fractional Brownian Motions, Fractional Noises and Applications," SIAM Review , Vol. 10, No. 4 (Oct., 1968), pp. 422-437
- [102] B. B. Mandelbrot, *Fractals and Scaling in Finance*, Springer, 1997.
- [103] J. Marro and R. Dickman, *Nonequilibrium Phase Transitions in Lattice Models*, Cambridge University Press, 1999.
- [104] R. Marculescu and P. Bogdan, "The Chip Is the Network: Toward a Science of Network-on-Chip Design," Foundations and Trends in Electronic Design Automation, vol. 2, no. 4, pp. 371-461, March, 2009.
- [105] J. M. Mendel, "Tutorial on higher-order statistics (spectra) in signal processing and system theory: Theoretical results and some applications," IEEE Proc., vol. 79, no. 3, pp. 278-305, Mar. 1991.
- [106] V. Mendez, S. Fedotov and W. Horsthemke, *Reaction-Transport Systems: Mesoscopic Foundations, Fronts, and Spatial Instabilities*, Springer - Science, Jun 1, 2010.
- [107] R. Metzler and J. Klafter, "The Random Walk's Guide to Anomalous Diffusion: A Fractional Dynamics Approach", Physics Reports, 339, pp. 1-77, 2000.
- [108] E. Miluzzo, N. D. Lane, K. Fodor, R. Peterson, H. Lu, M. Musolesi, S. B. Eisenman, X. Zheng and A. T. Campbell, "Sensing Meets Mobile Social Networks: The Design, Implementation

-
- and Evaluation of the CenceMe Application,” in Proceedings of the 6th ACM Conference on Embedded Network Sensor Systems, Raleigh, USA, 2008.
- [109] P. Mohan, V. N. Padmanabhan and R. Ramjee, “Nericell: Rich Monitoring of Road and Traffic Conditions Using Mobile SmartPhones,” in Proceedings of the 6th ACM Conference on Embedded Network Sensor Systems, Raleigh, USA, November 5-7, 2008.
- [110] E. W. Montroll and B. J. West, “On an Enriched Collection of Stochastic Processes,” in E.W. Montroll and J. Lebowitz (Eds.), *Fluctuation Phenomena*, Studies in Statistical Mechanics, vol. II, 1979.
- [111] J. D. Murray, *Mathematical Biology*, 3rd ed., Springer-Verlag Berlin Heidelberg, 2002.
- [112] National Science Foundation, “Cyber-physical systems,” Available online at <http://www.nsf.gov/publications/pubsumm.jsp?odskey=nsf08611>
- [113] M. E. J. Newman and D. J. Watts, *The Structure and Dynamics of Networks*, Princeton University Press, 2006.
- [114] K. Nowka, S. Nassif, K. Agarwal, “Characterization and Design for Variability and Reliability,” IEEE Custom Integrated Circuits Conference, pp.341-346, 21-24 September 2008.
- [115] L. Nottale, *Fractal Space-Time and Microphysics: Towards a Theory of Scale Relativity*, World Scientific, 1993.
- [116] G. Odor, *Universality in Nonequilibrium Lattice Systems: Theoretical Foundations*, World Scientific, Science, 2008
- [117] U. Y. Ogras, P. Bogdan and Radu Marculescu, “An Analytical Approach for Network-on-Chip Performance Analysis,” IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems, vol.29, no.12, pp.2001-2013, December 2010.
- [118] K. B. Oldham, J. Spanier, *Fractional Calculus: Theory of Differentiation and Integration to Arbitrary Order*, Academic, 2006.
- [119] A. Oustaloup, *La Dérivation Non Entière: Théorie, Synthèse et Applications*, Hermès, 1995.
- [120] K. Park and W. Willinger, *Self-similar Network Traffic and Performance Evaluation*, John Wiley & Sons, New York, 2000.
- [121] V. Paxson and S. Floyd, “Wide-area traffic: The failure of Poisson modeling,” IEEE Trans. Networking, vol. 3, no. 3, pp. 226-244, Jun. 1995.
- [122] C. E. M. Pearce, “The Exact Solution of the General Stochastic Rumor,” Journal of Mathematical and Computer Modelling, Vol. 31, 2000.

-
- [123] A. Pekalski and K. Sznajd-Weron, *Anomalous Diffusion: From Basics To Applications*, Springer - Science, 1999.
- [124] C.-K. Peng, S. V. Buldyrev, S. Havlin, M. Simons, H. E. Stanley, and A. L. Goldberger, “Mosaic organization of DNA nucleotides,” Phys. Rev. E 49, pp. 1685-1689, 1994.
- [125] L. Pietronero, *Fractals’ Physical Origin and Properties*, Plenum Press, 1989.
- [126] PhysioNet - The Research Resource for Complex Physiologic Signals, Available online: <http://www.physionet.org/>
- [127] I. Podlubny, L. Dorcak, and I. Kostial, “On Fractional Derivatives, Fractional-Order Dynamic Systems and $PI^{\lambda}D^{\mu}$ -controllers,” in Proc. IEEE Conf. Decision Control, San Diego, CA, pp. 4985-4990, 1997.
- [128] A.D. Polyanin and V.F. Zaitsev, *Handbook of Nonlinear Partial Differential Equations*, Chapman & Hall CRC, 2004.
- [129] Y. Qian, Z. Lu, and W. Dou, “Analysis of worst-case delay bounds for on-chip packet-switching networks,” IEEE Trans. Comput.-Aided Design Integr. Circuits Syst., vol. 29, no. 5, pp. 802-815, May 2010.
- [130] R. Rajkumar, I. Lee, L. Sha, and J. Stankovic, “Cyber Physical Systems: The Next Computing Revolution,” in Proceedings of the 47th ACM Design Automation Conference (DAC), Anaheim, June 2010.
- [131] P. Ranganathan, “From Microprocessors to Nanostores: Rethinking Data-Centric Systems,” IEEE Computer, vol.44, no.1, Jan. 2011.
- [132] P. Révész, Random walk in random and non-random environments, World Scientific, 2005.
- [133] I. Richardson, H.264 and MPEG-4 Video Compression: Video Coding for Next-generation Multimedia, John Wiley and Sons Press, 2003.
- [134] A. Richardella, P. Roushan, S. Mack, B. Zhou, D. A. Huse, D. D. Awschalom and A. Yazdani, “Visualizing Critical Correlations Near the Metal-Insulator Transition in Ga_{1-x}Mn_xAs,” Science, 327, pp. 665-669, 5 February 2010.
- [135] T. Sakurai and R. Newton, “Alpha-Power Law MOSFET Model and its Applications to CMOS Inverter Delay and Other Formulas,” IEEE Journal of Solid-State Circuits, vol. 25, no. 2, April 1990.
- [136] S. G. Samko, A. A. Kilbas, O. I. Marichev, *Fractional Integrals and Derivatives: Theory and Applications*, Gordon & Breach, 1993.
- [137] A. Schadschneider, T. Pöschel, R. Kühne, M. Schreckenberg and D. E. Wolf, Traffic and Granular Flow, Springer, 2007.

-
- [138] A. Scherrer, A. Fraboulet, and T. Risset, “Long-range dependence and on-chip processor traffic,” *Microprocess. Microsyst.* vol. 33, no. 1, pp. 72-80, Feb. 2009.
- [139] M. Schulz, *Statistical Physics and Economics: Concepts, Tools, and Applications*, Springer - Science, 2003.
- [140] F. Schweitzer and J. D. Farmer, *Brownian Agents and Active Particles: Collective Dynamics in the Natural and Social Sciences*, Springer Science, 2007.
- [141] J. I. Seiferas, Machine-Independent Complexity Theory, In: van Leeuwen J. (ed.), *Handbook of Theoretical Computer Science*, vol. A, pp. 165-186, Elsevier Science Publishers 1998.
- [142] Semiconductor Association. The International Technology Roadmap for Semiconductors (ITRS), 2010.
- [143] M. Sitti, “Miniature devices: Voyage of the microrobots,” *Nature*, 458, pp. 1121-1122, April 30th, 2009.
- [144] D. Sornette, *Critical Phenomena in Natural Sciences: Chaos, Fractals, Self-Organization, and Disorder*, Springer - Science, 2004.
- [145] V. Soteriou, H. Wang, and L.-S. Peh, “A statistical traffic model for on-chip interconnection networks,” in Proc. 14th Int. Symp. Modeling Anal. Simul. Comput. Telecommun. Syst., pp. 104-116, Sep. 2006.
- [146] Standard Performance Evaluation Corporation, Inc. SPECweb99 Benchmark. <http://www.spec.org/osg/web99>.
- [147] J. A. Stankovic, I. Lee, A. Mok and R. Rajkumar, “Opportunities and Obligations for Physical Computing Systems,” *Computer* 38, 11, pp. 23-31, November 2005.
- [148] H. E. Stanley and N. Ostrowsky, *Random Fluctuations and Pattern Growth: Experiments and Models*, Proceedings of the NATO Advanced Study Institute on Random Fluctuations and Pattern Growth, Cargese, Corsica, France, July 1988.
- [149] P. Tabuada, “Cyber-Physical Systems,” National Science Foundation Workshop on Cyber-Physical Systems, October, 2006.
- [150] H. Takayasu and A. Y. Tretyakov, “Extinction, Survival, and Dynamical Phase Transition of Branching Annihilating Random Walk,” *Phys. Rev. Lett.*, vol. 68, no. 20, pp. 3060-3063, 1992.
- [151] V.E. Tarasov, *Fractional Dynamics: Applications of Fractional Calculus to Dynamics of Particles, Fields and Media*, Springer, 2010.
- [152] A.M. Turing, “On Computable Numbers, with an Application to the Entscheidungsproblem,” *Proceedings of the London Mathematical Society*, 2, 42, pp. 230-265, 1936, “Correction,” 43, pp. 544-546, 1937.

-
- [153] V. Tuzlukov, *Signal Processing Noise*, CRC Press, 2002.
- [154] G. Varatkar and R. Marculescu, “On-chip traffic modeling and synthesis for MPEG-2 video applications,” IEEE Trans. Very Large Scale Integr. Syst., vol. 12, no. 1, pp. 108-119, Jan. 2004.
- [155] T. Vicsek, *Fractal growth phenomena*, World Scientific, 1992.
- [156] J. Warnock, “Circuit Design Challenges at the 14nm Technology Node,” *48th ACM/EDAC/IEEE Design Automation Conference (DAC)*, pp. 464-467, 5-9 June 2011
- [157] L. Wasserman, *All of Statistics: A Concise Course in Statistical Inference*, Springer Texts in Statistics, 2004.
- [158] R. Weinstock, *Calculus of Variations: With Applications to Physics and Engineering*, Dover, Mathematics, 1974.
- [159] J.J. Welser, G.I. Bourianoff, V.V. zhirnov, and R.K. Cavin III, “The Quest for Next Information Processing Technology,” Journal of NanoparticleResearch, vol. 10, pp. 1-10, 2008.
- [160] T. F. Wenisch, R. E. Wunderlich, M. Ferdman, A. Ailamaki, B. Falsafi, and J. C. Hoe, “Simflex: Statistical Sampling of Computer System Simulation,” IEEE Micro, 26(4), 2006.
- [161] B. J. West, M. Bologna and P. Grigolini, *Physics of Fractal Operators*, Springer, 2003.
- [162] B. J. West, “Fractal Physiology, Complexity and the Fractional Calculus,” in *Fractal, Diffusion, and Relaxation in Disordered Complex Systems*, edited by Y. P. Kalmykov and W. T. Coffey, John Wiley & Sons, 2006.
- [163] S.C. Woo, M. Ohara, E. Torrie, J.P. Singh, and A. Gupta, “The SPLASH-2 Programs: Characterization and Methodological Considerations,” in Proceedings of the 22nd Intl. Symp. on Computer Architecture, Santa Margherita Ligure, Italy, pp. 24-36, Jun. 1995.
- [164] C. Yen-Kuang, J. Chhugani, P. Dubey, C.J. Hughes, K. Daehyun S. Kumar, V. W. Lee, A.D. Nguyen, M. Smelyanskiy, “Convergence of Recognition, Mining, and Synthesis Workloads and Its Implications,” in Proceedings of IEEE, vol.96, no.5, pp.790-807, May 2008.
- [165] G. M. Zaslavsky, *Hamiltonian Chaos and Fractional Dynamics*, Oxford University Press - Mathematics, 2005.
- [166] G. M. Zaslavsky, *The Physics of Chaos in Hamiltonian Systems*, Imperial College Press, 2007.
- [167] J. J. Zebrowski and R. Baranowski, “Type I Intermittency in Nonstationary Systems - Models and Human Heart Rate Variability,” Physica A, 336, pp. 74-83, 2004.
- [168] V.V. Zhirnov, R.K., III Cavin,J.A. Hutchby and G.I. Bourianoff, “Limits to Binary Logic Switch Scaling - A Gedanken Model,” in Proceedings of the IEEE, vol.91, no.11, pp. 1934-1939, Nov 2003.