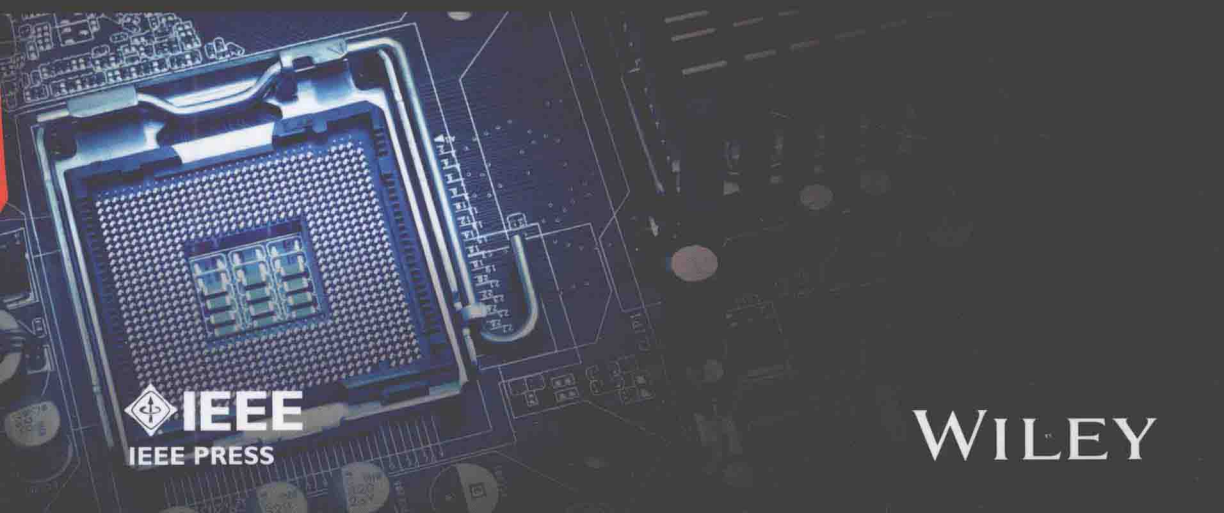


Modeling and Optimization of Parallel and Distributed Embedded Systems

Arslan Munir • Ann Gordon-Ross • Sanjay Ranka



 **IEEE**
IEEE PRESS

WILEY

MODELING AND OPTIMIZATION OF PARALLEL AND DISTRIBUTED EMBEDDED SYSTEMS

Arslan Munir

University of Nevada, Reno, USA

Ann Gordon-Ross

University of Florida, Gainesville, USA

Sanjay Ranka

University of Florida, Gainesville, USA

WILEY

 **IEEE**
IEEE PRESS

This edition first published 2016

© 2016 John Wiley & Sons Ltd

Registered office

John Wiley & Sons Ltd, The Atrium, Southern Gate, Chichester, West Sussex, PO19 8SQ, United Kingdom

For details of our global editorial offices, for customer services and for information about how to apply for permission to reuse the copyright material in this book please see our website at www.wiley.com.

The right of the author to be identified as the author of this work has been asserted in accordance with the Copyright, Designs and Patents Act 1988.

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording or otherwise, except as permitted by the UK Copyright, Designs and Patents Act 1988, without the prior permission of the publisher.

Wiley also publishes its books in a variety of electronic formats. Some content that appears in print may not be available in electronic books.

Designations used by companies to distinguish their products are often claimed as trademarks. All brand names and product names used in this book are trade names, service marks, trademarks or registered trademarks of their respective owners. The publisher is not associated with any product or vendor mentioned in this book.

Limit of Liability/Disclaimer of Warranty: While the publisher and author have used their best efforts in preparing this book, they make no representations or warranties with respect to the accuracy or completeness of the contents of this book and specifically disclaim any implied warranties of merchantability or fitness for a particular purpose. It is sold on the understanding that the publisher is not engaged in rendering professional services and neither the publisher nor the author shall be liable for damages arising herefrom. If professional advice or other expert assistance is required, the services of a competent professional should be sought.

Library of Congress Cataloging-in-Publication Data applied for:

ISBN: 9781119086413

A catalogue record for this book is available from the British Library.

Cover Image: PaulPaladin, Plus69/Getty

Set in 10/12pt, TimesLTStd by SPi Global, Chennai, India.

Printed and bound in Singapore by Markono Print Media Pte Ltd

Preface

Advancements in silicon technology, micro-electro-mechanical systems (MEMS), wireless communications, computer networking, and digital electronics have led to the proliferation of embedded systems in a plethora of application domains (e.g., industrial and home automation, automotive, space, medical, and defense). To meet the diverse application requirements of these application domains, novel trends have emerged in embedded systems. One such trend is networking single-unit embedded systems to form a multiple-unit embedded system, also referred to as a *distributed embedded system*. Given the collective computing capabilities of the single-unit embedded systems, the distributed embedded system enables more sophisticated applications of greater value as compared to an isolated single-unit embedded system. An emerging trend is to connect these distributed embedded systems via a wireless network instead of a bulky, wired networking infrastructure. Another emerging trend in embedded systems is to leverage multicore/manycore architectures to meet the continuously increasing performance demands of many application domains (e.g., medical imaging, mobile signal processing). Both single-unit and distributed embedded systems can leverage multicore architectures for attaining high performance and energy efficiency. Since processing is done in parallel with multicore-based embedded systems, these systems are being termed as *parallel embedded systems*. The burgeoning of multicore architectures in embedded systems induces parallel computing into the embedded domain, which was previously used predominantly in the supercomputing domain only. In some applications, parallel embedded systems are networked together to form *parallel and distributed embedded systems*. For both parallel and distributed embedded systems, modeling and optimization at various design levels (e.g., verification, simulation, analysis) are of paramount significance. Considering the short time-to-market for many embedded systems, often embedded system designers resort to modeling approaches for the evaluation of design alternatives in terms of performance, power, reliability, and/or scalability.

About This Book

Embedded computers have advanced well beyond the early days of 8-bit microcontrollers. Contemporary embedded computers are organized into multiprocessors that execute millions of lines of code in real time and at very low power levels. This book targets *parallel and distributed embedded systems*, which have been enabled by technological advances in silicon technology, MEMS, wireless communications, computer networking, and digital electronics.

These parallel and distributed embedded systems have applications in various domains, such as military and defense, medical, automotive, and unmanned autonomous vehicles.

This book discusses parallel and distributed embedded systems with a main focus on three *design metrics*:

- Performance
- Power
- Dependability

Often design metrics have conflicting resource requirements, and the interplay between these design metrics presents interesting research challenges.

The emphasis of this book is on modeling and optimization of emerging parallel and distributed embedded systems with respect to these design metrics.¹ To illustrate the modeling and optimization of distributed embedded systems, we present our work on modeling and optimization of embedded sensor nodes in embedded wireless sensor networks (EWSNs). This book discusses optimization strategies employed in EWSNs at different design levels to meet application requirements, such as lifetime, throughput, and reliability. To illustrate modeling in distributed embedded systems, we discuss an application metrics estimation model that estimates high-level application metrics (e.g., lifetime, throughput) from low-level sensor node tunable parameters (e.g., processor voltage and frequency) and the sensor node's hardware internals (e.g., transceiver voltage, transceiver receive current). We discuss Markov Decision Process (MDP) for dynamic optimization of embedded sensor nodes in EWSNs to more closely adhere to application requirements. MDP is suitable for the dynamic optimization of the embedded sensor nodes because of MDP's inherent ability to perform dynamic decision making.

To enable in situ autonomous EWSN dynamic optimizations, we examine an online EWSN optimization methodology that extends static design time *parameter tuning*. Parameter tuning is the process of determining appropriate *parameter values* (e.g., processor voltage, processor frequency, sensing frequency) that meet application requirements. Static tuning optimizes an EWSN at deployment time and remains fixed for the EWSN's lifetime. Whereas static optimizations are suitable for stable/predictable applications, static optimizations are inflexible and do not adapt to changing application requirements and environmental stimuli. Our proposed methodology is advantageous over static design time parameter tuning because our methodology enables the embedded sensor node to automatically adapt to actual changing environmental stimuli, resulting in closer adherence to application requirements. Since many application designers are nonexperts (e.g., agriculturist, biologists) and lack sufficient expertise for parameter tuning, autonomous parameter tuning methodologies may alleviate many of these design challenges. Our methodology is more amenable to nonexpert application designers and requires no application designer effort after initial EWSN deployment. We develop lightweight (low computational and memory resources) online algorithms for the optimization methodology. These lightweight algorithms are crucial for embedded sensor nodes considering limited processing, storage, and energy resources of embedded sensor nodes in distributed EWSNs. We further discuss modeling and analysis of fault detection and fault tolerance (FT) in EWSNs.

¹ Major portion of this book stems from the Ph.D. dissertation of Dr. Arslan Munir at the University of Florida, Gainesville, USA.

This book elaborates on various high-performance and energy-efficient techniques at the architecture, middleware, and software levels for parallel multicore-based embedded systems. To elucidate the modeling of parallel embedded systems, this book discusses in detail our proposed queueing theoretic approach for modeling the performance of multicore embedded systems. This modeling technique enables quick and inexpensive architectural evaluation in terms of both design time and resources as compared to developing and/or using existing multicore simulators and running benchmarks on these simulators. Based on a preliminary evaluation using our model, architecture designers can run targeted benchmarks to further verify the performance characteristics of selected multicore architectures (i.e., our queueing theory-based model facilitates early design space pruning).

The book elucidates performance aspects of parallel multicore-based embedded systems by performance analysis of contemporary multicore embedded architectures: symmetric multiprocessors (SMPs) and tiled multicore architectures (TMAs). We compare the performance of SMPs and TMAs based on a parallelized information fusion application, a Gaussian elimination, and embarrassingly parallel benchmarks. We provide a quantitative comparison of these two architectures based on the calculations of various device metrics (e.g., computational density (CD), CD per watt (CD/W), internal memory bandwidth (IMB), and external memory bandwidth (EMB)). Although a quantitative comparison provides a high-level evaluation of the computational capabilities of the architectures, our work provides deeper insights based on parallelized benchmark-driven evaluation.

To illustrate parallel and distributed embedded systems with a unifying example, we discuss an architecture for heterogeneous hierarchical multicore embedded wireless sensor networks (MCEWSNs). We elaborate several compute-intensive tasks, such as information fusion, encryption, network coding, and software-defined radio, which will benefit in particular from the increased computation power offered by multicore embedded sensor nodes. We characterize and discuss various application domains for MCEWSNs. The book concludes with the research challenges and future research directions for parallel and distributed embedded systems.

Highlights

This book focuses on modeling, analysis, and optimization of parallel and distributed embedded systems. To illustrate the modeling and optimization of distributed embedded systems, we focus on modeling and optimization of distributed EWSNs. Specifically, we target dynamic optimization methodologies based on the sensor node tunable parameter value settings for EWSNs. We elaborate additional modeling issues in parallel embedded systems, which have come into existence owing to the burgeoning multicore/manycore revolution, by our performance modeling of multicore parallel embedded systems.

The highlights of this book include the following:

Overview of Parallel and Distributed Embedded Systems

The book gives an overview of parallel and distributed embedded systems and narrates the application of these embedded systems in various application domains.

Multicore-Based EWSNs—An Example of Parallel and Distributed Embedded Systems:

We deliberate on the feasibility and application of multicore parallel architectures as processing units in distributed embedded sensor nodes. We propose a MCEWSN architecture based on multicore embedded sensor nodes, which serves as an example of parallel and distributed embedded systems. Furthermore, we summarize the multicore initiative in EWSNs by academia and industry.

Modeling of Parallel and Distributed Embedded Systems

The book includes various chapters illustrating modeling of parallel and distributed embedded systems.

Application Metrics Estimation Model: The book discusses an application metrics estimation model that estimates high-level application metrics (e.g., lifetime, throughput) from low-level embedded sensor node tunable parameters and the embedded sensor node's hardware internals (e.g., processor voltage and frequency). Our dynamic optimization methodologies for embedded sensor nodes leverage this estimation model while comparing different operating states for optimization purposes.

Modeling and Analysis of Fault Detection and Fault Tolerance: The book illustrates the modeling of reliable embedded systems via modeling of fault detection and FT in EWSNs. We propose an FT sensor node model consisting of duplex sensors (i.e., one active sensor and one inactive spare sensor), which exploits the synergy of fault detection and FT.

Queueing Theoretic Modeling of Multicore-Based Parallel Embedded Architectures: The book proposes a queueing theory-based modeling technique for evaluating multicore parallel embedded architectures. The modeling technique would enable quick and inexpensive architectural evaluation in terms of both design time and resources as compared to developing and/or using existing multicore simulators and running benchmarks on these simulators.

Optimization of Parallel and Distributed Embedded Systems

The book includes various chapters illustrating optimizations of parallel and distributed embedded systems.

Optimization Approaches in Distributed Embedded Systems: The book discusses various optimization approaches in distributed embedded systems focusing on distributed EWSNs.

Performance and Energy Optimizations for Parallel Embedded Systems: We elaborate on performance and energy optimizations for parallel embedded systems. Although the literature discusses high-performance parallel computing for supercomputers, there exists little discussion on high-performance energy-efficient parallel embedded computing.

Dynamic Optimization Methodologies: We explore dynamic optimization methodologies for distributed embedded systems with EWSNs as an example. First, we propose an MDP-based dynamic optimization methodology for embedded sensor nodes. MDP is suitable for EWSN dynamic optimization because of MDP's inherent ability to perform dynamic decision making. We then present online algorithms for dynamic optimization of embedded sensor nodes in distributed EWSNs. Lightweight (low computational

and memory resources) online algorithms for design space exploration are crucial for embedded sensor nodes considering the limited processing, storage, and energy resources of embedded sensor nodes. We discuss a lightweight dynamic optimization methodology for embedded sensor nodes that intelligently selects appropriate initial tunable parameter value settings by evaluating the application's unique requirements, the relative importance of these requirements with respect to each other, and the magnitude in which each parameter affects each requirement.

Analysis and Optimization of Parallel Embedded Systems: The book conducts a performance analysis of parallel embedded systems focusing on SMPs and TMAs—an architectural innovation in the multicore technology—based on parallelized benchmarks. We identify key architecture and software optimizations to attain high performance from parallel embedded systems focusing on TMAs. We discuss the performance optimizations on a single tile (processor core) as well as on parallel performance optimizations, such as application decomposition, cache locality, tile locality, memory balancing, and horizontal communication for TMAs. We elaborate on compiler-based optimizations that are applicable to TMAs, such as function inlining, loop unrolling, and feedback-based optimizations. We present a case study using optimized dense matrix multiplication algorithms for TMAs to experimentally demonstrate the performance and performance per watt optimizations on TMAs.

Intended Audience

The book is mainly aimed as a reference book for students and researchers in embedded systems. The book's main target audience consists of senior undergraduate students, graduate students, and researchers involved in embedded systems research. Some chapters of the book contain advanced material suitable for senior undergraduate students, graduate students, and researchers in computer engineering or computer science; however, the book also contains several introductory chapters that do not require any deep knowledge of embedded systems to follow the material. The book can be considered as a mix of introductory and advanced materials on embedded systems.

Organization of the Book

This book is organized into three parts:

- Overview
- Modeling
- Optimization

The “Overview” part provides an introduction to embedded systems, modeling, and optimization of parallel and distributed embedded systems. The “Modeling” and “Optimization” parts illustrate modeling and optimization issues, respectively, through EWSNs and multicore. Following is the description of the organization of the chapters into three parts of the book.

The “Overview” part consists of Chapters 1 and 2. Chapter 1 gives an overview of parallel and distributed embedded systems. Chapter 2 proposes an MCEWSN architecture based on multicore embedded sensor nodes. The MCEWSN architecture serves as an example for

parallel and distributed embedded systems. The chapter then summarizes the multicore and parallel computing initiatives in EWSNs by academia and industry.

The “Modeling” part consists of Chapters 3–5. Chapter 3 presents an application metrics estimation model for distributed EWSN. Modeling and analysis of fault detection and fault tolerance in EWSNs are presented in Chapter 4. Chapter 5 discusses a novel, queueing theory-based modeling technique for evaluating multicore embedded architectures for parallel embedded systems.

The “Optimization” part consists of Chapters 6–12. Chapter 6 discusses various optimization approaches in distributed embedded systems focusing on distributed EWSNs. Various performance and energy optimizations for parallel multicore embedded systems are discussed in Chapter 7. The MDP-based dynamic optimization methodology for embedded sensor nodes in distributed EWSNs is presented in Chapter 8. Online algorithms for embedded sensor node’s tunable parameter-based dynamic optimization are presented in Chapter 9. Chapter 10 presents a lightweight dynamic optimization methodology for embedded sensor nodes in distributed EWSNs. Chapter 11 evaluates performance of parallel embedded systems leveraging SMPs and TMAs based on parallelized benchmarks. Chapter 12 identifies key architecture and software optimizations to attain high performance from parallel embedded systems and illustrates these optimizations via a case study of dense matrix multiplication on TMAs. Finally, Chapter 13 concludes this book.

Acknowledgment

This work was supported in part by the National Science Foundation (CNS-0953447 and CNS-0905308). Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation. This work was also supported in part by the US Department of Energy, National Nuclear Security Administration, and Advanced Simulation and Computing Program, as a Cooperative Agreement under the Predictive Science Academic Alliance Program, under Contract No. DE-NA0002378.

Contents

Preface	xv
Acknowledgment	xxi
Part I OVERVIEW	
1 Introduction	3
1.1 Embedded Systems Applications	6
1.1.1 <i>Cyber-Physical Systems</i>	6
1.1.2 <i>Space</i>	6
1.1.3 <i>Medical</i>	7
1.1.4 <i>Automotive</i>	8
1.2 Characteristics of Embedded Systems Applications	9
1.2.1 <i>Throughput-Intensive</i>	9
1.2.2 <i>Thermal-Constrained</i>	9
1.2.3 <i>Reliability-Constrained</i>	10
1.2.4 <i>Real-Time</i>	10
1.2.5 <i>Parallel and Distributed</i>	10
1.3 Embedded Systems—Hardware and Software	11
1.3.1 <i>Embedded Systems Hardware</i>	11
1.3.2 <i>Embedded Systems Software</i>	14
1.4 Modeling—An Integral Part of the Embedded Systems Design Flow	15
1.4.1 <i>Modeling Objectives</i>	16
1.4.2 <i>Modeling Paradigms</i>	18
1.4.3 <i>Strategies for Integration of Modeling Paradigms</i>	20
1.5 Optimization in Embedded Systems	21
1.5.1 <i>Optimization of Embedded Systems Design Metrics</i>	23
1.5.2 <i>Multiobjective Optimization</i>	26
1.6 Chapter Summary	27

2	Multicore-Based EWSNs—An Example of Parallel and Distributed Embedded Systems	29
2.1	Multicore Embedded Wireless Sensor Network Architecture	31
2.2	Multicore Embedded Sensor Node Architecture	33
2.2.1	<i>Sensing Unit</i>	34
2.2.2	<i>Processing Unit</i>	34
2.2.3	<i>Storage Unit</i>	34
2.2.4	<i>Communication Unit</i>	35
2.2.5	<i>Power Unit</i>	35
2.2.6	<i>Actuator Unit</i>	35
2.2.7	<i>Location Finding Unit</i>	36
2.3	Compute-Intensive Tasks Motivating the Emergence of MCEWSNs	36
2.3.1	<i>Information Fusion</i>	36
2.3.2	<i>Encryption</i>	38
2.3.3	<i>Network Coding</i>	38
2.3.4	<i>Software-Defined Radio (SDR)</i>	38
2.4	MCEWSN Application Domains	38
2.4.1	<i>Wireless Video Sensor Networks (WVSNs)</i>	39
2.4.2	<i>Wireless Multimedia Sensor Networks (WMSNs)</i>	39
2.4.3	<i>Satellite-Based Wireless Sensor Networks (SBWSN)</i>	40
2.4.4	<i>Space Shuttle Sensor Networks (3SN)</i>	41
2.4.5	<i>Aerial–Terrestrial Hybrid Sensor Networks (ATHSNs)</i>	42
2.4.6	<i>Fault-Tolerant (FT) Sensor Networks</i>	43
2.5	Multicore Embedded Sensor Nodes	43
2.5.1	<i>InstraNode</i>	43
2.5.2	<i>Mars Rover Prototype Mote</i>	43
2.5.3	<i>Satellite-Based Sensor Node (SBSN)</i>	44
2.5.4	<i>Multi-CPU-Based Sensor Node Prototype</i>	44
2.5.5	<i>Smart Camera Mote</i>	44
2.6	Research Challenges and Future Research Directions	45
2.7	Chapter Summary	47

Part II MODELING

3	An Application Metrics Estimation Model for Embedded Wireless Sensor Networks	51
3.1	Application Metrics Estimation Model	52
3.1.1	<i>Lifetime Estimation</i>	53
3.1.2	<i>Throughput Estimation</i>	56
3.1.3	<i>Reliability Estimation</i>	57
3.1.4	<i>Models Validation</i>	57
3.2	Experimental Results	58
3.2.1	<i>Experimental Setup</i>	58
3.2.2	<i>Results</i>	59
3.3	Chapter Summary	61

4	Modeling and Analysis of Fault Detection and Fault Tolerance in Embedded Wireless Sensor Networks	63
4.1	Related Work	67
4.1.1	<i>Fault Detection</i>	67
4.1.2	<i>Fault Tolerance</i>	68
4.1.3	<i>WSN Reliability Modeling</i>	69
4.2	Fault Diagnosis in WSNs	70
4.2.1	<i>Sensor Faults</i>	70
4.2.2	<i>Taxonomy for Fault Diagnosis Techniques</i>	72
4.3	Distributed Fault Detection Algorithms	74
4.3.1	<i>Fault Detection Algorithm 1: The Chen Algorithm</i>	74
4.3.2	<i>Fault Detection Algorithm 2: The Ding Algorithm</i>	76
4.4	Fault-Tolerant Markov Models	77
4.4.1	<i>Fault-Tolerance Parameters</i>	77
4.4.2	<i>Fault-Tolerant Sensor Node Model</i>	79
4.4.3	<i>Fault-Tolerant WSN Cluster Model</i>	81
4.4.4	<i>Fault-Tolerant WSN Model</i>	83
4.5	Simulation of Distributed Fault Detection Algorithms	85
4.5.1	<i>Using ns-2 to Simulate Faulty Sensors</i>	85
4.5.2	<i>Experimental Setup for Simulated Data</i>	86
4.5.3	<i>Experiments Using Real-World Data</i>	86
4.6	Numerical Results	91
4.6.1	<i>Experimental Setup</i>	91
4.6.2	<i>Reliability and MTTF for an NFT and an FT Sensor Node</i>	91
4.6.3	<i>Reliability and MTTF for an NFT and an FT WSN Cluster</i>	95
4.6.4	<i>Reliability and MTTF for an NFT and an FT WSN</i>	98
4.7	Research Challenges and Future Research Directions	101
4.7.1	<i>Accurate Fault Detection</i>	101
4.7.2	<i>Benchmarks for Comparing Fault Detection Algorithms</i>	101
4.7.3	<i>Energy-Efficient Fault Detection and Tolerance</i>	101
4.7.4	<i>Machine-Learning-Inspired Fault Detection</i>	102
4.7.5	<i>FT in Multimedia Sensor Networks</i>	102
4.7.6	<i>Security</i>	102
4.7.7	<i>WSN Design and Tuning for Reliability</i>	104
4.7.8	<i>Novel WSN Architectures</i>	104
4.8	Chapter Summary	105
5	A Queuing Theoretic Approach for Performance Evaluation of Low-Power Multicore-Based Parallel Embedded Systems	107
5.1	Related Work	110
5.2	Queuing Network Modeling of Multicore Embedded Architectures	112
5.2.1	<i>Queuing Network Terminology</i>	112
5.2.2	<i>Modeling Approach</i>	113
5.2.3	<i>Assumptions</i>	119
5.3	Queuing Network Model Validation	120
5.3.1	<i>Theoretical Validation</i>	120

5.3.2	<i>Validation with a Multicore Simulator</i>	120
5.3.3	<i>Speedup</i>	124
5.4	Queueing Theoretic Model Insights	125
5.4.1	<i>Model Setup</i>	125
5.4.2	<i>The Effects of Cache Miss Rates on Performance</i>	129
5.4.3	<i>The Effects of Workloads on Performance</i>	132
5.4.4	<i>Performance per Watt and Performance per Unit Area Computations</i>	135
5.5	Chapter Summary	139

Part III OPTIMIZATION

6	Optimization Approaches in Distributed Embedded Wireless Sensor Networks	143
6.1	Architecture-Level Optimizations	144
6.2	Sensor Node Component-Level Optimizations	146
6.2.1	<i>Sensing Unit</i>	146
6.2.2	<i>Processing Unit</i>	148
6.2.3	<i>Transceiver Unit</i>	148
6.2.4	<i>Storage Unit</i>	148
6.2.5	<i>Actuator Unit</i>	148
6.2.6	<i>Location Finding Unit</i>	149
6.2.7	<i>Power Unit</i>	149
6.3	Data Link-Level Medium Access Control Optimizations	149
6.3.1	<i>Load Balancing and Throughput Optimizations</i>	149
6.3.2	<i>Power/Energy Optimizations</i>	150
6.4	Network-Level Data Dissemination and Routing Protocol Optimizations	152
6.4.1	<i>Query Dissemination Optimizations</i>	152
6.4.2	<i>Real-Time Constrained Optimizations</i>	154
6.4.3	<i>Network Topology Optimizations</i>	154
6.4.4	<i>Resource-Adaptive Optimizations</i>	154
6.5	Operating System-Level Optimizations	155
6.5.1	<i>Event-Driven Optimizations</i>	155
6.5.2	<i>Dynamic Power Management</i>	155
6.5.3	<i>Fault Tolerance</i>	155
6.6	Dynamic Optimizations	156
6.6.1	<i>Dynamic Voltage and Frequency Scaling</i>	156
6.6.2	<i>Software-Based Dynamic Optimizations</i>	156
6.6.3	<i>Dynamic Network Reprogramming</i>	157
6.7	Chapter Summary	157
7	High-Performance Energy-Efficient Multicore-Based Parallel Embedded Computing	159
7.1	Characteristics of Embedded Systems Applications	163
7.1.1	<i>Throughput-Intensive</i>	163
7.1.2	<i>Thermal-Constrained</i>	165

7.1.3	<i>Reliability-Constrained</i>	165
7.1.4	<i>Real-Time</i>	165
7.1.5	<i>Parallel and Distributed</i>	165
7.2	Architectural Approaches	166
7.2.1	<i>Core Layout</i>	166
7.2.2	<i>Memory Design</i>	168
7.2.3	<i>Interconnection Network</i>	170
7.2.4	<i>Reduction Techniques</i>	172
7.3	Hardware-Assisted Middleware Approaches	173
7.3.1	<i>Dynamic Voltage and Frequency Scaling</i>	174
7.3.2	<i>Advanced Configuration and Power Interface</i>	174
7.3.3	<i>Gating Techniques</i>	175
7.3.4	<i>Threading Techniques</i>	176
7.3.5	<i>Energy Monitoring and Management</i>	177
7.3.6	<i>Dynamic Thermal Management</i>	178
7.3.7	<i>Dependable Techniques</i>	179
7.4	Software Approaches	180
7.4.1	<i>Data Forwarding</i>	180
7.4.2	<i>Load Distribution</i>	180
7.5	High-Performance Energy-Efficient Multicore Processors	182
7.5.1	<i>ARM11 MPCore</i>	183
7.5.2	<i>ARM Cortex A-9 MPCore</i>	184
7.5.3	<i>MPC8572E PowerQUICC III</i>	184
7.5.4	<i>Tilera TILEPro64 and TILE-Gx</i>	184
7.5.5	<i>AMD Opteron Processor</i>	185
7.5.6	<i>Intel Xeon Processor</i>	185
7.5.7	<i>Intel Sandy Bridge Processor</i>	185
7.5.8	<i>Graphics Processing Units</i>	186
7.6	Challenges and Future Research Directions	186
7.7	Chapter Summary	189
8	An MDP-Based Dynamic Optimization Methodology for Embedded Wireless Sensor Networks	191
8.1	Related Work	193
8.2	MDP-Based Tuning Overview	195
8.2.1	<i>MDP-Based Tuning Methodology for Embedded Wireless Sensor Networks</i>	195
8.2.2	<i>MDP Overview with Respect to Embedded Wireless Sensor Networks</i>	197
8.3	Application-Specific Embedded Sensor Node Tuning Formulation as an MDP	200
8.3.1	<i>State Space</i>	200
8.3.2	<i>Decision Epochs and Actions</i>	200
8.3.3	<i>State Dynamics</i>	201
8.3.4	<i>Policy and Performance Criterion</i>	201
8.3.5	<i>Reward Function</i>	202
8.3.6	<i>Optimality Equation</i>	204
8.3.7	<i>Policy Iteration Algorithm</i>	205

8.4	Implementation Guidelines and Complexity	205
	8.4.1 <i>Implementation Guidelines</i>	205
	8.4.2 <i>Computational Complexity</i>	206
	8.4.3 <i>Data Memory Analysis</i>	207
8.5	Model Extensions	207
8.6	Numerical Results	210
	8.6.1 <i>Fixed Heuristic Policies for Performance Comparisons</i>	210
	8.6.2 <i>MDP Specifications</i>	210
	8.6.3 <i>Results for a Security/Defense System Application</i>	213
	8.6.4 <i>Results for a Healthcare Application</i>	216
	8.6.5 <i>Results for an Ambient Conditions Monitoring Application</i>	220
	8.6.6 <i>Sensitivity Analysis</i>	222
	8.6.7 <i>Number of Iterations for Convergence</i>	223
8.7	Chapter Summary	223
9	Online Algorithms for Dynamic Optimization of Embedded Wireless Sensor Networks	225
9.1	Related Work	227
9.2	Dynamic Optimization Methodology	228
	9.2.1 <i>Methodology Overview</i>	228
	9.2.2 <i>State Space</i>	229
	9.2.3 <i>Objective Function</i>	229
	9.2.4 <i>Online Optimization Algorithms</i>	230
9.3	Experimental Results	233
	9.3.1 <i>Experimental Setup</i>	233
	9.3.2 <i>Results</i>	235
9.4	Chapter Summary	239
10	A Lightweight Dynamic Optimization Methodology for Embedded Wireless Sensor Networks	241
10.1	Related Work	243
10.2	Dynamic Optimization Methodology	244
	10.2.1 <i>Overview</i>	244
	10.2.2 <i>State Space</i>	246
	10.2.3 <i>Optimization Objection Function</i>	246
10.3	Algorithms for Dynamic Optimization Methodology	248
	10.3.1 <i>Initial Tunable Parameter Value Settings and Exploration Order</i>	248
	10.3.2 <i>Parameter Arrangement</i>	249
	10.3.3 <i>Online Optimization Algorithm</i>	251
	10.3.4 <i>Computational Complexity</i>	252
10.4	Experimental Results	252
	10.4.1 <i>Experimental Setup</i>	253

10.4.2	<i>Results</i>	255
10.5	Chapter Summary	266
11	Parallelized Benchmark-Driven Performance Evaluation of Symmetric Multiprocessors and Tiled Multicore Architectures for Parallel Embedded Systems	269
11.1	Related Work	271
11.2	Multicore Architectures and Benchmarks	272
11.2.1	<i>Multicore Architectures</i>	272
11.2.2	<i>Benchmark Applications and Kernels</i>	273
11.3	Parallel Computing Device Metrics	275
11.4	Results	277
11.4.1	<i>Quantitative Comparison of SMPs and TMAs</i>	277
11.4.2	<i>Benchmark-Driven Results for SMPs</i>	278
11.4.3	<i>Benchmark-Driven Results for TMAs</i>	280
11.4.4	<i>Comparison of SMPs and TMAs</i>	282
11.5	Chapter Summary	285
12	High-Performance Optimizations on Tiled Manycore Embedded Systems: A Matrix Multiplication Case Study	287
12.1	Related Work	290
12.1.1	<i>Performance Analysis and Optimization</i>	290
12.1.2	<i>Parallelized MM Algorithms</i>	290
12.1.3	<i>Cache Blocking</i>	291
12.1.4	<i>Tiled Manycore Architectures</i>	292
12.2	Tiled Manycore Architecture (TMA) Overview	293
12.2.1	<i>Intel's TeraFLOPS Research Chip</i>	294
12.2.2	<i>IBM's Cyclops-64 (C64)</i>	296
12.2.3	<i>Tilera's TILEPro64</i>	297
12.2.4	<i>Tilera's TILE64</i>	300
12.3	Parallel Computing Metrics and Matrix Multiplication (MM) Case Study	301
12.3.1	<i>Parallel Computing Metrics for TMAs</i>	301
12.3.2	<i>Matrix Multiplication (MM) Case Study</i>	302
12.4	Matrix Multiplication Algorithms' Code Snippets for Tilera's TILEPro64	303
12.4.1	<i>Serial Non-blocked Matrix Multiplication Algorithm</i>	303
12.4.2	<i>Serial Blocked Matrix Multiplication Algorithm</i>	304
12.4.3	<i>Parallel Blocked Matrix Multiplication Algorithm</i>	307
12.4.4	<i>Parallel Blocked Cannon's Algorithm for Matrix Multiplication</i>	309
12.5	Performance Optimization on a Manycore Architecture	314
12.5.1	<i>Performance Optimization on a Single Tile</i>	314
12.5.2	<i>Parallel Performance Optimizations</i>	315
12.5.3	<i>Compiler-Based Optimizations</i>	319