Preface

Computing with dynamic reconfiguration has developed into a mature area, with a rich collection of techniques, results, and algorithms. A dynamically reconfigurable architecture (or simply a reconfigurable architecture) typically consists of a large number of computing elements connected by a reconfigurable communication medium. By dynamically restructuring connections between the computing elements, these architectures admit extremely fast solutions to several computational problems. The interaction between computation and the communication medium permits novel techniques not possible on a fixed-connection network.

This book spans the large body of work on dynamically reconfigurable architectures and algorithms. It is not an exhaustive collection of results in this area, rather, it provides a comprehensive view of dynamic reconfiguration by emphasizing fundamental techniques, issues, and algorithms. The presentation includes a wide repertoire of topics, starting from a historical perspective on early reconfigurable systems, ranging across a wide variety of results and techniques for reconfigurable models, examining more recent developments such as optical models and run-time reconfiguration (RTR), and finally touching on an approach to implementing a dynamically reconfigurable model.

Researchers have developed algorithms on a number of reconfigurable architectures, generally similar, though differing in details. One aim of this book is to present these algorithms in the setting of a single computational platform, the reconfigurable mesh (R-Mesh). The R-Mesh possesses the basic features of a majority of other architectures and is sufficient to run most algorithms. This structure can help the reader relate results and understand techniques without the haze of details on just what is and is not allowed from one model to the next.

For algorithms that require additional features beyond those of the R-Mesh, we highlight the features and reasons for their use. For example, having directional buses permits an R-Mesh to solve the directed graph reachability problem in constant time, which is not known to be (and not likely to be) possible for an R-Mesh with undirected buses. In describing this algorithm, we pinpoint just where directed buses permit the algorithm to succeed and undirected buses would fail.

Although most of the book uses the R-Mesh as the primary vehicle of expression, a substantial portion deals with the relationships between the R-Mesh and other models of computation, both reconfigurable and traditional (Chapters 9 and 10). The simulations integral to developing these relationships also provide generic methods to translate algorithms between models.

The book is addressed to researchers, graduate students, and system designers. To the researcher, it offers an extensive digest of topics ranging from basic techniques and algorithms to theoretical limits of computing on reconfigurable architectures. To the system designer, it provides a comprehensive reference to tools and techniques in the area. In particular, Part IV of the book deals with optical models and Field Programmable Gate Arrays (FPGAs), providing a bridge between theory and practice.

The book contains over 380 problems, ranging in difficulty from those meant to reinforce concepts to those meant to fill gaps in the presentation to challenging questions meant to provoke further thought. The book features a list of figures, a rich set of bibliographic notes at the end of each chapter, and an extensive bibliography. The book also includes a comprehensive index with topics listed under multiple categories. For topics spanning several pages, page numbers of key ideas are in bold.

Organization of Book

The book comprises four parts. Part I (Chapters 1–3) provides a first look at reconfiguration. It includes introductory material, describing the overall nature of reconfiguration, various models and architectures, important issues, and fundamental algorithmic techniques. Part II (Chapters 4–7) deals with algorithms on reconfigurable architectures for a variety of problems. Part III (Chapters 8 and 9) describes self and mutual simulations for several reconfigurable models, placing their computational capabilities relative to traditional parallel models of computation and complexity classes. Part IV (Chapters 10 and 11) touches on capturing, in the models themselves, the effect of practical constraints, providing a bridge between theory and practice. Each chapter is reasonably self-contained and includes a set of exercises and bibliographic notes.

The presentation in this book is suitable for a graduate level course and only presupposes basic ideas in parallel computing. Such a course could include (in addition to Chapters 1–3), Chapters 4–8 (for an emphasis on algorithms), Chapters 4–6, 8, 9 (for a more theoretical flavor), portions of Chapters 4–6 along with Chapters 8, 10, 11 (to stress aspects with more bearing on implementability). This book could also serve as a companion text to most graduate courses on parallel computing.

Chapter 1: Principles and Issues

This chapter introduces the idea of dynamic reconfiguration and reviews important considerations in reconfigurable architectures. It provides a first look at the R-Mesh, the model used for most of the book.

Chapter 2: The Reconfigurable Mesh: A Primer

This chapter details the R-Mesh model and uses it to describe several techniques commonly employed in algorithms for reconfigurable architectures. It also develops a palette of fundamental algorithms that find use as building blocks in subsequent chapters.

Chapter 3: Models of Reconfiguration

This chapter provides an overview of other models of reconfiguration with examples of their use. These models include restrictions and enhancements of the R-Mesh, other bus-based models, optical models, and field programmable gate arrays. It describes relationships among these models based on considerations of computing power and implementability.

Chapter 4: Arithmetic on the R-Mesh

Unlike conventional computing platforms, resource bounds for even simple arithmetic operations on reconfigurable architectures depend on the size and representations of the inputs. This chapter addresses these issues and describes algorithms for a variety of problems, including techniques for fast addition, multiplication, and matrix multiplication.

Chapter 5: Sorting and Selection

This chapter deals with problems on totally ordered sets and includes techniques for selection, area-optimal sorting, and speed-efficiency trade-offs.

Chapter 6: Graph Algorithms

Methods for embedding graphs in reconfigurable models are described in the context of list ranking and graph connectivity. Along with other techniques such as tree traversal, rooting, and labeling, these techniques illustrate how methods developed on non-reconfigurable models can be translated to constant-time algorithms on reconfigurable architectures.

Chapter 7: Computational Geometry & Image Processing
This chapter applies the methods developed in preceding chapters to solve problems such as convex hull, Voronoi diagrams, histogramming, and quadtree generation.

Chapter 8: Model and Algorithmic Scalability
Normally, algorithm design does not consider the relationship between problem size and the size of the available machine. This chapter deals with issues that arise from a mismatch between the problem and machine sizes, and introduces methods to cope with them.

Chapter 9: Computational Complexity of Reconfiguration
This chapter compares the computational "powers" of different reconfigurable models and places "reconfigurable complexity classes" in relation to conventional Turing machine, PRAM, and circuit complexity classes.

Chapter 10: Optical Reconfigurable Models
This chapter describes reconfigurable architectures that employ fiber-optic buses. An optical bus offers a useful pipelining technique that permits moving large amounts of information among processors despite a small bisection width. The chapter introduces models, describes algorithms and techniques, and presents complexity results.

Chapter 11: Run-Time Reconfiguration
This chapter details run-time reconfiguration techniques for Field Programmable Gate Arrays (FPGAs) and touches upon the relationships between FPGA-type and R-Mesh-type platforms. Towards this end, it presents an approach to implementing R-Mesh algorithms on an FPGA-type environment.

Acknowledgments. We are grateful to the National Science Foundation for its support of our research on dynamic reconfiguration; numerous results in this book and much of the insight that formed the basis of our presentation resulted from this research. Our thanks also go to Hossam ElGindy, University of New South Wales, for his suggestions on organizing Chapter 2. Our thanks go to numerous students for their constructive criticisms of a preliminary draft of this book. Most importantly, this book would not have been possible without the patience and support of our families. We dedicate this book to them.

R. Vaidyanathan

J. L. Trahan