This book is dedicated to the memory of Charlton Hall Amos.

Preface

DNA computation has emerged in the last ten years as an exciting new research field at the intersection (and, some would say, frontiers) of computer science, biology, engineering, and mathematics. Although anticipated by Feynman as long ago as the 1950s [59], the notion of performing computations at a molecular level was only realized in 1994, with Adleman's seminal work [3] on computing with DNA. Since then the field has blossomed rapidly, with significant theoretical and experimental results being reported regularly.

Several books [120, 39] have described various aspects of DNA computation, but this is, to the author's best knowledge, the first to bring together descriptions of both theoretical and experimental results. The target audience is intentionally broad, including students as well as experienced researchers. We expect that users of the book will have some background in either computer science, mathematics, engineering, or the life sciences. The intention is that this book be used as a tutorial guide for newcomers to the field as well as a reference text for people already working in this fascinating area. To this end, we include two self-contained tutorial chapters (1 and 2), which convey only those aspects of computer science and biology that are required to understand the subsequent material.

We now describe in detail the structure of the book. An introduction places what follows in context, before we motivate the work to be presented by emphasizing some of the reasons for choosing DNA over other candidate computational substrates. One reason for choosing to work with DNA is the size and variety of the molecular "tool box" available to us. Since the discovery of the structure of DNA, a wide range of biological tools have been developed to facilitate and ease genetic experimentation. Molecular biologists have available to them a much more diverse set of methods for the manipulation of DNA than they have for any other molecule. It is possible that similar techniques could, in the future, become available for molecules other than DNA, but existing methods have the advantage of being ubiquitous, tried, and tested. We discuss these methods in more detail in Chap. 1.

In Chap. 2 we introduce some fundamental concepts in theoretical computer science and define what it means to "compute." We introduce the notion of a *computational problem*, and explain why some problems are fundamentally more difficult than others. This motivates Chap. 3, in which we show how models of molecular computation may be constructed. We describe several such formalizations that appeared soon after [3], and discuss their computational power.

One reason for the current interest in using DNA for computations is the massive parallelism inherent in laboratory operations on this particular molecule. When we perform a step in an experiment (say, adding an enzyme to a test-tube), the operation is performed *in parallel* on every molecule in the tube (in reality, reaction dynamics affect the overall efficiency of this process, but we discount this for the sake of clarity). If we consider that a single drop of solution can contain trillions of DNA molecules, the potential for *massive* parallelism is apparent. However, as we shall see, it is important to harness this parallelism correctly if we are to make significant progress. In Chap. 4 we discuss complexity issues in DNA computing, and outline the shortcomings of early models in the light of these.

By describing attempts to characterize the complexity of molecular algorithms in Chap. 4, we motivate a discussion of *feasible* and *efficient* models of DNA computation. We describe several models that attempt to drive the field closer to the so-called "killer application," the application of DNA-based computers that would establish their superiority in a particular domain.

Chapter 5 is concerned with physical realizations of some of the models outlined in Chap. 3. We describe several laboratory experiments and the lessons to be derived from the results obtained. We also describe exciting "late-breaking" laboratory results that appeared too recently to deal with in detail.

In Chap. 6 we discuss recent work that has focused attention on the potential for performing computations in vivo, as opposed to in vitro, as has previously always been the case. We examine the inner workings of the cell from a computational perspective, and describe recent theoretical and experimental developments.

Core work reported in this monograph was performed by the author as a doctoral student under the supervision of Alan Gibbons. I owe much to, and am grateful for, Alan's wisdom, generosity, patience, and humor. I thank Grzegorz Rozenberg for offering me the opportunity to publish this work in the Natural Computing series, and for his friendship and guidance. Many other colleagues have contributed to the work presented, especially Paul E. Dunne, David Hodgson, Gerald Owenson, and Steve Wilson. Chris Tofts was the external examiner of my doctoral thesis, and offered useful advice on content and presentation. I thank Cheryl Sutton for her help with final typesetting, and the assistance provided by Dr. Hans Wössner, Ingeborg Mayer, and Ronan Nugent at Springer, and by Christina Brückner at LE-Tex, is also gratefully

acknowledged. Finally, I would like to express my gratitude for continued support from my wife, Justine Ashby.

Exeter

Martyn Amos
December, 2004

http://www.springer.com/978-3-540-65773-6

Theoretical and Experimental DNA Computation

Amos, M.

2005, XIII, 173 p., Hardcover
ISBN: 978-3-540-65773-6