

Preface

The interactions between different formal models and biology have a long-standing successful history. Results have been produced on both sides of this process while continuously new models are considered and new biological territories are uncovered.

On one hand, biological systems, as cells, have been recognised as complex systems requesting various models to simulate, explain, and verify their multiple properties. A very popular notion of complex systems is that of a very large number of simple and identical elements interacting to produce complex emergent behaviour. Unlike complex systems of simple elements, in which functions emerge from the properties of the network they form rather than from any specific elements, functions of biological systems rely on a combination of the network and the specific elements involved. Molecular biology has uncovered a great deal of biological facts, each one being very important in isolation, but various biological entities (cells, tissues, organs, organisms, colonies, societies) reveal very complex interactions. This level of aggregation, generally called systems biology, requires a combination of approaches, from experiments to abstract formal models able to manage the huge complexity of the biological systems to accurately represent and simulate them. In order to capture the complexity and the dynamicity of these systems, various models have been developed ranging from abstract mathematical-based models (either continuous or discrete approaches) to different specific programming paradigms or sophisticated software systems.

In the same time, natural sciences, and especially biology, represented a rich source of modelling paradigms. Well-defined areas of artificial intelligence (genetic algorithms, neural networks), mathematics, and theoretical computer science (L systems, DNA computing) are massively influenced by the behaviour of various biological entities and phenomena. In the last decades or so, new

emerging fields of so-called “natural computing” identify new (unconventional) computational paradigms in different forms. There are attempts to define and investigate new mathematical or theoretical models inspired by nature, as well as investigations into defining programming paradigms that implement computational approaches suggested by biochemical phenomena. Especially since Adleman’s experiment, these investigations received a new perspective. One hopes that global system-level behaviour may be translated into interactions of a myriad of components with simple behaviour and limited computing and communication capabilities that are able to express and solve, via various optimisations, complex problems otherwise hard to approach.

AIMS OF THE BOOK

Primarily, the book aims to:

- Overview a number of areas of natural computing (P systems, networks of evolutionary processors) by revealing the problems that are mostly researched and pointing towards future developments.
- Analyse and suggest various interactions between different approaches of the same biological phenomena.
- Discuss variants of classical concepts, such as dynamical systems, in rather unconventional settings (formal theory of languages, programming paradigms using evolutionary developments).
- Cover (nearly) all aspects of modelling (formal specifications, implementation, formal verification, simulations, predictions).
- Reveal the advantages of using hybrid complex unconventional models in order to express complex interactions occurring in complex biological entities (bacteria) or systems with a dynamic changeable structure.

AUDIENCE OF THE BOOK

The book is written to be used by a number of distinct audiences. It is mainly written for researchers developing new models, especially with a nature-inspired flavour, looking at new, surprising mathematical or theoretical properties of these models and also pursuing research in systems biology or computational biology. The book is also suitable for researchers and practitioners in the area of programming languages and paradigms who are interested in new programming concepts inspired from biology, with a bias towards developmental structures and topologies. Academics may find the book

useful as a textbook presenting some advanced topics in computer science or a source of problems and research-led topics for projects or postgraduate courses.

OUTLINE OF THE BOOK

The book is organized into nine chapters.

Chapter 1 (Gheorghe Păun) presents an overview on membrane computing, a branch of natural computing whose initial goal was to abstract computing models from the structure and the functioning of the living cells. The research was initiated about five years ago (at the end of 1998), and since that time the subject has been developed significantly from a mathematical point of view. The basic types of results of this research concern computability power (in comparison with the standard Turing machines and their restrictions) and efficiency (the possibility to solve computationally hard problems, typically NP-complete problems, in a feasible time, almost polynomial). However, membrane computing has recently become attractive also as a framework for devising models of biological phenomena, with the tendency to provide tools for modelling the cell itself, not only the local processes. Chapter 1 surveys the basic elements of membrane computing, somewhat in its “historical” evolution, from biology to computer science and mathematics and back to biology. The presentation is informal, without any technical detail and an invitation to membrane computing intended to acquaint the nonmathematician reader with the main directions of research of the domain, the type of central results, and the possible lines of future development, including the possible interest of the biologist looking for discrete algorithmic tools for modelling cell phenomena.

Chapter 2 (Vincenzo Manca, Giuditta Franco, and Giuseppe Scollo) introduces some of the classical dynamics concepts in the basic mathematical setting of state transition systems where time and space are completely discrete and no structure is assumed on the state’s space. Interesting relationships between attractors and recurrence are identified, and some features of chaos are expressed in simple, set theoretic terms. String dynamics is proposed as a unifying concept for dynamical systems arising from discrete models of computation together with illustrative examples. The relevance of state transition systems and string dynamics is discussed from the perspective of molecular computing.

Chapter 3 (Lila Kari, Elena Losseva, and Petr Sosík) is a survey that examines the question of managing errors that arise in DNA-based computa-

tion. Due to the inaccuracy of biochemical reactions, the experimental implementation of a DNA computation may lead to incorrectly calculated results. This chapter looks at different methods that can assist in reduction of such occurrences. The solutions to the problem of erroneous bio-computations are presented from the perspective of computer science techniques. Three main aspects of dealing with errors are covered: software simulations, algorithmic approaches, and theoretical methods. The objective of this survey is to explain how these tools can reduce errors associated with DNA computing.

Chapter 4 (Carlos Martín-Vide and Victor Mitrana) surveys, in a systematic and uniform way, the main results regarding different computational aspects of hybrid networks of evolutionary processors viewed both as generating and accepting devices, as well as of solving problems with these mechanisms. The chapter first explains how generating hybrid networks of evolutionary processors is computationally complete. The same computational power is reached by accepting hybrid networks of evolutionary processors. It is then defined as a computational complexity class of accepting hybrid networks of evolutionary processors and proven that this class equals the classical class NP. The chapter also presents a few NP-complete problems and recalls how they can be solved in linear time by accepting networks of evolutionary processors with linearly bounded resources (nodes, rules, symbols). Finally, the chapter discusses some possible directions for further research.

Chapter 5 (Andrés Cordon Franco, Miguel Angel Gutiérrez Naranjo, Mario J. Pérez Jiménez, and Agustín Riscos Nuñez) is devoted to the study of numerical NP-complete problems in the framework of cellular systems with membranes, also called P systems (Păun, 1998). The chapter presents *efficient* solutions to the Subset Sum and the Knapsack problems. These solutions are obtained via families of P systems with the capability of generating an exponential working space in polynomial time. A simulation tool for P systems, written in Prolog, is also described. As an illustration of the use of this tool, the chapter includes a session in the Prolog simulator implementing an algorithm to solve a subset sum or knapsack problem.

Chapter 6 (Jean-Louis Giavitto and Olivier Michel) focuses on a model inspired by biological development, both at the molecular and cellular levels. Such biological processes are particularly interesting for computer science because the dynamic organization emerges from many decentralized and local interactions that occur concurrently at several time and space scales. Thus, they provide a source of inspiration to solve various problems related to mobility, distributed systems, open systems, and others. The fundamental mechanisms of biological development are now understood as changes within a com-

plex dynamical system. This chapter advocates that these fundamental mechanisms, although mainly developed in a continuous framework, can be rephrased in a discrete setting relying on the notion of rewriting in a topological setting. The discrete formulation is as formal as the continuous one, enables the simulation, and opens a way to the systematic study of the behavioral properties of the biological systems. Directly inspired from these developmental processes, it is presented as an experimental programming language called MGS. MGS is dedicated to the modelling and simulation of dynamical systems with dynamical structures. The chapter illustrates the basic notions of MGS through several algorithmic examples and by sketching various biological models.

Chapter 7 (Ray C. Paton, Richard Gregory, Costas Vlachos, JohnW. Palmer, Jon R. Saunders, and Q. H. Wu) describes two approaches to individual-based modelling that are based on bacterial evolution and bacterial ecologies. Some history of the individual-based modelling approach is presented and contrasted to traditional methods. Two related models of bacterial evolution are then discussed in some detail. The first model consists of populations of bacterial cells, each containing a genome, or gene products developed through transcription cascade and mutation. As a result, this model contains multiple time scales and is very fine-grained. The second model employs a coarser-grained, agent-based architecture, designed to explore the evolvability of adaptive behavioural strategies in artificial bacterial ecologies. The organisms in this approach are represented by mutating learning classifier systems. Finally, the subject of computability on parallel machines and clusters is applied to these models, with the aim of making them efficiently scalable to the point of being biologically realistic by containing sufficient numbers of complex individuals.

Chapter 8 (Gabriel Ciobanu) describes a model of the molecular networks by using a system of communicating automata as a dynamic structure and discrete event system, providing interesting theoretical results. This formal model provides a detailed approach of the biological system, and its implementation is able to handle large amounts of data. This model is applied to a T cell signalling network. A T cell shows a hierarchical organization depending on various factors. Some mechanisms are still unresolved, including contribution of each signalling pathway to each response type. The software tool produced is used to simulate and analyze T cell behaviour. The simulation reflects quite faithfully the process of T cell activation and T cell responses. This increases the confidence to use this model and its implementation both as a descriptive and prescriptive tool. The interactions that govern T cell behaviour are simulated and analyzed, providing statistical correlations according to software experiments, together with new insights on signalling networks that trig-

ger immunological responses. The software tool allows users to systematically perturb and monitor the components of a T cell molecular network, capturing relevant biological information to immunology.

Chapter 9 (Petros Kefalas, George Eleftherakis, Mike Holcombe, and Ioanna Stamatopoulou) presents a formal method, namely X-machines, used to specify, verify, and test individual agents. Multi-agent systems are highly dynamic since the agents' abilities and the system configuration often changes over time. In some ways, such multi-agent systems seem to behave like biological processes; new agents appear in the system, some others cease to exist, and communication between agents changes. One of the challenges of multi-agent systems is to attempt to formally model their dynamic configuration. Utilized concepts from biological processes can identify and define a set of operations that are able to reconfigure a multi-agent system. This chapter presents an example of these concepts, in which a biology-inspired system is incrementally built in order to meet our objective.