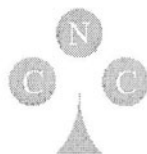


Natural Computing Series

Series Editors: G. Rozenberg (Managing)
Th. Bäck A.E. Eiben J.N. Kok H.P. Spaink
Leiden Center for Natural Computing



Advisory Board: S. Amari G. Brassard M. Conrad
K.A. De Jong C.C.A.M. Gielen T. Head L. Kari
L. Landweber T. Martinetz Z. Michalewicz M.C. Mozer
E. Oja G. Păun J. Reif H. Rubin A. Salomaa M. Schoenauer
H.-P. Schwefel D. Whitley E. Winfree J.M. Zurada

Springer

Berlin

Heidelberg

New York

Barcelona

Hong Kong

London

Milan

Paris

Tokyo

Gheorghe Păun

Membrane Computing

An Introduction

With 37 Figures and 5 Tables



Springer

Gheorghe Păun
Institute of Mathematics
of the Romanian Academy
P.O.Box 1-764
70700 Bucuresti, Romania
and
Rovira i Virgili University
Pl. Imperial Tarraco, 1
43005 Tarragona, Spain
gpaun@imar.ro, gp@astor.urv.es
www.imar.ro/~gpaun

Series Editors
G. Rozenberg (Managing Editor)
Th. Bäck, A.E. Eiben, J.N. Kok, H.P. Spaink
Leiden Center for Natural Computing
Leiden University
Niels Bohrweg 1
2333 CA Leiden
The Netherlands
rozenber@cs.leidenuniv.nl

Library of Congress Cataloging-in-Publication Data

Păun, Gheorghe, 1950-
Membrane computing: an introduction/Gheorghe Păun.
p. cm. – (Natural computing series)
Includes bibliographical references and index.
ISBN 3540436014 (hc: alk. paper)
I. Molecular computers. I. Title. II. Series.

QA 76.887 .P39 2002

621.39'1–dc21

2002075730

ACM Computing Classification (1998): F.1.1, F.4.2, F.4.3

ISSN 1619-7127

ISBN 3-540-43601-4 Springer-Verlag Berlin Heidelberg New York

This work is subject to copyright. All rights are reserved, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilm or in any other way, and storage in data banks. Duplication of this publication or parts thereof is permitted only under the provisions of the German Copyright Law of September 9, 1965, in its current version, and permission for use must always be obtained from Springer-Verlag. Violations are liable for prosecution under the German Copyright Law.

Springer-Verlag Berlin Heidelberg New York,
a member of BertelsmannSpringer Science+Business Media GmbH
<http://www.springer.de>

© Springer-Verlag Berlin Heidelberg 2002

The use of general descriptive names, trademarks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.

Cover Design: KünkelLopka, Heidelberg
Typesetting: Camera ready by the author
Printed on acid-free paper SPIN 10877938 45/3142PS – 5 4 3 2 1 0

**To my wife Anișoara
(Ana de la Argeș)**

Preface

Membrane computing is a branch of natural computing which abstracts from the structure and the functioning of living cells. In the basic model, the membrane systems – also called P systems – are distributed parallel computing devices, processing multisets of objects, synchronously, in the compartments delimited by a membrane structure. The objects, which correspond to chemicals evolving in the compartments of a cell, can also pass through membranes. The membranes form a hierarchical structure – they can be dissolved, divided, created, and their permeability can be modified. A sequence of transitions between configurations of a system forms a computation. The result of a halting computation is the number of objects present at the end of the computation in a specified membrane, called the output membrane. The objects can also have a structure of their own that can be described by strings over a given alphabet of basic molecules – then the result of a computation is a set of strings. An important version of membrane systems deals with membranes arranged not in a hierarchical structure (which mathematically corresponds to a tree), but in a tissue-like structure (which mathematically corresponds to a graph).

This computing model (more accurately, this class of computing models) was proposed at the end of 1998, and, in less than four years since then (until the time of completing the manuscript of this book), more than 180 papers have been written, as well as three PhD theses (with several more in preparation now) on the subject. Hence the task of presenting membrane computing in a monograph is by no means a simple one. In particular, new notions are currently introduced, new results are proved and old results are improved; the focus of research may still change depending on various criteria (ranging from mathematical appeal to practical attractiveness), and therefore, the task of choosing central (lasting) notions and results for this monograph was really difficult.

This is why this book is only an introduction to membrane computing, with a subjective selection of basic notions and (classes of) results. The book is only a snapshot of developments in a vivid research area, and it is meant to be an invitation to witness and to participate in this research. Hopefully, further “generations” of this book will be made both possible and necessary by the reader’s contributions to membrane computing – such contributions

are very much encouraged by the style of the book: it is as self-contained as possible from biological and computer science points of view, with many open problems and research topics explicitly formulated. The problems are numbered, from Q1 to Q39; a table at the end of the book gives the pages where these problems are stated.

Another list at the end of the book contains the universality results proved or only stated in the text; the goal of this list is to provide a clear overview of universality results, and in particular of the role of various parameters (such as, for example, the number of membranes) in obtaining universality.

The reader interested in further details, or in the current developments in membrane computing, is invited to visit the web page at the address <http://bioinformatics.bio.disco.unimib.it/psystems>.

*

This book has grown continuously during the last four years, in collaboration with the many friends from many countries with whom I have had the privilege to work. I will only name them, hopefully without too many omissions: I. Ardelean, F. Arroyo, A. Atanasiu, A. Baranda, D. Besozzi, P. Bottoni, J. Castellanos, G. Ciobanu, E. Csuhaj-Varjú, J. Dassow, C. Ferretti, R. Freund, P. Frisco, T. Head, H.J. Hoogeboom, M. Ito, J. Kelemen, V. Manca, S. Marcus, M. Margenstern, C. Martín-Vide, A. Mateescu, G. Mauri, V. Mitran, J. Pazos, A. Păun, M. Pérez-Jiménez, A. Rodríguez-Patón, Y. Rogozhin, G. Rozenberg, Y. Sakakibara, A. Salomaa, F. Sancho-Caparrini, Y. Suzuki, H. Tanaka, T. Yokomori, S. Yu, C. Zandron. Several of them have read (parts of) the manuscript of the present book, making valuable observations.

To all of them, the warmest thanks and indebtedness.

Of great help also have been the participants to the annual workshops on membrane computing held in Curtea de Argeş, Romania, since 2000.

Special thanks are due to Claudio Zandron, the webmaster taking care of the Internet page mentioned above.

Thanks are offered in advance to readers who take up the present invitation to membrane computing, to send their comments about the text or, mainly, contributions to the contents of a possible future version of the book.

The writing of this book has benefited from the wonderful working conditions provided by the Research Group on Mathematical Linguistics, of Rovira i Virgili University, Tarragona, Spain, where I work as a researcher on the Ramon y Cajal programme of the Spanish Ministry of Research.

Last but not least, I should mention the pleasant and efficient collaboration with Springer-Verlag. Mrs Ingeborg Mayer deserves a special bow in this respect.

Contents

| | |
|--|-----|
| Preface | VII |
| 1. Introduction: Membrane Computing – What It Is and What It Is Not | 1 |
| 2. Prerequisites | 7 |
| 2.1 The Biological Membrane | 7 |
| 2.1.1 The Structure of the Plasma Membrane | 8 |
| 2.1.2 Trans-membrane Transport | 10 |
| 2.1.3 Cellular Division: Mitosis | 14 |
| 2.2 The Neuron | 15 |
| 2.3 Elements of Computability | 16 |
| 2.3.1 Basic Notions and Notations | 17 |
| 2.3.2 Operations with Strings and Languages | 18 |
| 2.3.3 Chomsky Grammars | 19 |
| 2.3.4 Characterizations, Necessary Conditions | 22 |
| 2.3.5 Lindenmayer Systems | 24 |
| 2.3.6 Finite Automata, Turing Machines | 26 |
| 2.3.7 Regulated Rewriting | 29 |
| 2.3.8 On the Difference Between CS and RE | 39 |
| 2.3.9 Universal Turing Machines and Type-0 Grammars | 40 |
| 2.3.10 Splicing, Insertion–Deletion, Context Adjoining | 42 |
| 2.3.11 Elements of Complexity | 45 |
| 2.3.12 Multisets | 49 |
| 2.4 Bibliographical Notes | 50 |
| 3. Membrane Systems with Symbol–Objects | 51 |
| 3.1 A Simple Class | 51 |
| 3.2 Two Examples | 55 |
| 3.3 The Power of the Simple Class | 58 |
| 3.4 Basic Extensions | 64 |
| 3.4.1 Dissolving a Membrane | 64 |
| 3.4.2 Priorities Among the Evolution Rules | 70 |
| 3.4.3 Two Further Examples | 71 |

| | | |
|-----------|--|------------|
| 3.4.4 | The Power of Priority | 74 |
| 3.4.5 | The Power of Synchronization | 77 |
| 3.5 | A Formal Definition | 85 |
| 3.6 | Further Extensions | 91 |
| 3.6.1 | Weak Target Commands | 91 |
| 3.6.2 | Controlling the Permeability of Membranes | 93 |
| 3.6.3 | Communication Controlled by Concentration | 99 |
| 3.6.4 | Creating Rules During the Computation | 102 |
| 3.6.5 | Using Promoters/Inhibitors | 104 |
| 3.7 | Systems with External Output | 114 |
| 3.8 | Bibliographical Notes | 125 |
| 4. | Trading Evolution for Communication | 129 |
| 4.1 | Systems with Symport/Antiport | 130 |
| 4.2 | Computational Universality | 133 |
| 4.3 | Controls on the Use of Rules | 141 |
| 4.4 | Following the Traces of Objects | 144 |
| 4.5 | Systems with Carriers | 153 |
| 4.6 | Bibliographical Notes | 159 |
| 5. | Structuring the Objects | 161 |
| 5.1 | Rewriting Membrane Systems | 162 |
| 5.2 | Some Variants and Their Power | 180 |
| 5.2.1 | Rule Creation | 181 |
| 5.2.2 | Conditional Rewriting | 181 |
| 5.2.3 | Conditional Communication | 186 |
| 5.2.4 | Replicated Rewriting | 199 |
| 5.2.5 | Parallel Rewriting | 208 |
| 5.3 | Splicing Membrane Systems | 211 |
| 5.4 | Contextual Membrane Systems | 223 |
| 5.5 | Insertion–Deletion Membrane Systems | 226 |
| 5.6 | Bibliographical Notes | 231 |
| 6. | Networks of Membranes | 235 |
| 6.1 | The Splicing Case | 236 |
| 6.2 | Using Symport/Antiport Rules | 238 |
| 6.3 | Neural-like Networks of Membranes | 249 |
| 6.3.1 | Definitions and Examples | 249 |
| 6.3.2 | The Computational Power | 256 |
| 6.3.3 | The Computational Efficiency | 267 |
| 6.4 | Bibliographical Notes | 269 |

| | |
|---|-----|
| 7. Trading Space for Time | 271 |
| 7.1 Complexity Classes for Membrane Systems | 271 |
| 7.2 Using Membrane Division | 273 |
| 7.2.1 Solving SAT in Linear Time | 281 |
| 7.2.2 Solving the Hamiltonian Path Problem | 286 |
| 7.2.3 Using Cooperative Rules | 290 |
| 7.2.4 Is Membrane Division Necessary? | 298 |
| 7.3 Using Membrane Creation | 301 |
| 7.3.1 Solving SAT | 311 |
| 7.3.2 Solving HPP | 316 |
| 7.3.3 The Case of String-Objects | 318 |
| 7.4 Using String Replication | 321 |
| 7.5 Using Pre-computed Resources | 323 |
| 7.6 Bibliographical Notes | 327 |
| 8. Further Technical Results | 329 |
| 8.1 Decidability Results | 329 |
| 8.2 Unary Systems | 340 |
| 8.3 A Representation of Context-free Languages | 344 |
| 8.4 Valuating the String-Objects | 348 |
| 8.5 Systems with Enhanced Membrane Handling | 351 |
| 8.6 Brief Excursion Through the Literature | 354 |
| 8.6.1 Generalized Sequential Membrane Systems | 354 |
| 8.6.2 Bidimensional Objects | 357 |
| 8.6.3 Membrane Systems and Stream X-machines | 357 |
| 8.6.4 Membrane Systems and Ambient Calculus | 359 |
| 8.6.5 A Direct Construction of a Universal System | 361 |
| 8.6.6 Further Research Topics | 363 |
| 9. (Attempts to Get) Back to Reality | 367 |
| 9.1 Getting Closer to the Cell by Energy Accounting | 367 |
| 9.2 Getting Closer to the Cell by Gemmation | 373 |
| 9.3 Getting Closer to the Cell: Bilayer Membranes | 375 |
| 9.4 <i>In Silico</i> Implementations | 379 |
| 9.5 Artificial Life Applications | 384 |
| 9.6 A Simulation of Photosynthesis | 392 |
| Open Problems | 399 |
| Universality Results | 401 |
| References | 403 |
| Index | 417 |