

1. Record Nr.	UNINA9910488702403321
Autore	Zhang Gexiang
Titolo	Membrane computing models : implementations // Gexiang Zhang [and six others]
Pubbl/distr/stampa	Gateway East, Singapore : , : Springer, , [2021] ©2021
ISBN	981-16-1566-7
Descrizione fisica	1 online resource (291 pages)
Disciplina	006.38
Soggetti	Natural computation
Lingua di pubblicazione	Inglese
Formato	Materiale a stampa
Livello bibliografico	Monografia
Nota di contenuto	Intro -- Foreword -- Preface -- Acknowledgments -- Contents -- Acronyms -- 1 Introduction -- 1.1 Membrane Computing Overview -- 1.2 Software Implementation of P Systems -- 1.3 Hardware Implementation of P Systems -- 1.4 Challenges of P Systems Implementation -- 1.5 Concluding Remarks -- References -- 2 P Systems Implementation on P-Lingua Framework -- 2.1 Introduction -- 2.2 P-Lingua Language -- 2.2.1 P System Models -- 2.2.2 Membrane Structure -- 2.2.3 Initial Multisets -- 2.2.4 P System Rules -- 2.3 Simulation Algorithms -- 2.4 Membrane Computing Simulator (MeCoSim) -- 2.4.1 Primary goals -- 2.4.2 Main Functional Components -- 2.5 Conclusion -- References -- 3 Applications of Software Implementations of P Systems -- 3.1 Introduction -- 3.2 Automatic Design of Cell-Like P Systems with P-Lingua -- 3.2.1 Preliminaries -- 3.2.1.1 Alphabet and Multisets -- 3.2.1.2 Rooted Tree -- 3.2.1.3 Cell-Like P System/Transition P System -- 3.2.2 Automatic Design of P Systems with an Elitist Genetic Algorithm -- 3.2.2.1 Problem Statement -- 3.2.2.2 Design Method -- 3.2.3 Automatic Design of P Systems with a Permutation Penalty Genetic Algorithm -- 3.3 Automatic Design of Spiking Neural P Systems with P-Lingua -- 3.4 Modelling Real Ecosystems with MeCoSim -- 3.4.1 Problem Description -- 3.5 Robot Motion Planning -- 3.5.1 Problem Definition -- 3.5.2 Path Planning for Mobile Robots -- 3.5.3 Rapidly-Exploring Random Tree (RRT) Algorithm -- 3.6 Conclusion -- References -- 4 Infobiotics

Workbench: An In Silico Software Suite for Computational Systems Biology -- 4.1 Introduction -- 4.2 Stochastic P Systems -- 4.3 Software Description -- 4.3.1 Simulation -- 4.3.2 Verification -- 4.3.3 Optimization -- 4.4 Case Studies -- 4.4.1 Pulse generator -- 4.4.2 Repressilator -- 4.5 KPWorkbench: A Qualitative Analysis Tool -- 4.6 Next-Generation Infobiotics for Synthetic Biology. 4.7 Conclusion -- References -- 5 Molecular Physics and Chemistry in Membranes: The Java Environment for Nature-Inspired Approaches (JENA) -- 5.1 Introduction -- 5.2 JENA at a Glance and Its Descriptive Capacity -- 5.2.1 Atoms, Ions, Molecules, and Particles -- 5.2.2 Vessels and Delimiters -- 5.2.3 Brownian Motion and Thermodynamics -- 5.2.4 Chemical Reactions by Effective Collisions and by Spontaneous Decay -- 5.2.5 Applying External Forces -- 5.2.6 Active Membranes and Dynamical Delimiters -- 5.2.7 Simulation, Monitoring, Logging, and Analyses -- 5.3 JENA Source Code Design -- 5.4 Selection of JENA Case Studies -- 5.4.1 Chemical Lotka-Volterra Oscillator -- 5.4.2 Electrophoresis -- 5.4.3 Centrifugation -- 5.4.4 Neural Signal Transduction Across Synaptic Cleft -- 5.5 Conclusions and Prospectives -- References -- 6 P Systems Implementation on GPUs -- 6.1 Introduction -- 6.2 GPU Computing -- 6.2.1 The Graphics Processing Unit -- 6.2.2 CUDA Programming Model -- 6.2.3 GPU Architecture -- 6.2.4 Good Practices -- 6.3 Generic Simulations -- 6.3.1 Definition -- 6.3.2 Simulating P Systems with Active Membranes -- 6.3.2.1 Recognizer P Systems with Active Membranes -- 6.3.2.2 Simulation Algorithm -- 6.3.2.3 Sequential Simulator -- 6.3.2.4 Parallel Simulation on CUDA -- 6.3.2.5 Performance Comparative Analysis -- 6.3.3 Simulating Population Dynamics P Systems -- 6.3.3.1 Population Dynamics P Systems -- 6.3.3.2 Simulation Algorithm -- 6.3.3.3 Design of the Parallel Simulator -- 6.3.3.4 GPU Implementation of the DCBA Phases -- 6.3.3.5 Performance Results of the Simulator -- 6.4 Specific Simulations -- 6.4.1 Definition -- 6.4.2 Simulating a SAT Solution with Active Membrane P Systems -- 6.4.2.1 SAT Solution with Active Membranes -- 6.4.2.2 Sequential Simulator and Data Structures -- 6.4.2.3 Design of the GPU Simulator -- 6.4.2.4 Performance Analysis. 6.4.3 Simulating a SAT Solution with Tissue P Systems -- 6.4.3.1 Recognizer Tissue P System with Cell Division -- 6.4.3.2 SAT Solution with Tissue P Systems -- 6.4.3.3 Sequential Simulation and Data Structure -- 6.4.3.4 Design of the Parallel Simulator -- 6.4.3.5 Performance Analysis -- 6.5 Adaptive Simulations -- 6.5.1 Definition -- 6.5.2 Simulating Population Dynamics P Systems -- 6.5.2.1 Analysis of Performance Results -- 6.6 Conclusions -- References -- 7 P Systems Implementation on FPGA -- 7.1 Introduction -- 7.2 FPGA Hardware -- 7.3 Generalized Numerical P Systems (GNPS) -- 7.3.1 Formal Definition -- 7.3.2 Basic Variant -- 7.3.3 Historical Remarks -- 7.4 Implementing GNPS on FPGA -- 7.5 FPGA Implementations of Other Models of P Systems -- 7.5.1 Petreska and Teuscher Implementation -- 7.5.2 Nguyen Implementation -- 7.5.3 Quiros and Verlan Implementation -- 7.5.4 Comments -- 7.6 Discussion -- 7.7 Conclusion -- References -- 8 Applications of Hardware Implementation of P Systems -- 8.1 Introduction -- 8.2 Robot Membrane Controllers with FPGA Implementation -- 8.2.1 Numerical P Systems-Based Membrane Controllers on FPGA -- 8.2.2 Enzymatic Numerical P Systems (ENPS)-Based Membrane Controllers on FPGA -- 8.2.3 GNPS-Based Membrane Controllers on FPGA -- 8.3 Robot Path Planning with FPGA Implementation -- 8.3.1 RRT Algorithm -- 8.3.2 Arithmetic Units Design -- 8.3.3 Enzymatic Numerical P System Rapid-Exploring Random Tree Register Transfer Level (ENPS-RRT RTL) Model Design -- 8.3.4 ENPS-RRT on FPGA -- 8.4 Conclusion -- References --

