

004.42
A35

P.I. ANDON
A.YU. DOROSHENKO
K.A. ZHEREB
O.A. YATSENKO

**ALGEBRA-
ALGORITHMIC
MODELS
AND METHODS
OF PARALLEL
PROGRAMMING**

П.І. АНДОН, А.Ю. ДОРОШЕНКО, К.А. ЖЕРЕБ, О.А. ЯЦЕНКО

АЛГЕБРО - АЛГОРИТМІЧНІ МОДЕЛІ ТА МЕТОДИ ПАРАЛЕЛЬНОГО ПРОГРАМУВАННЯ

ПРОЕКТ
«УКРАЇНСЬКА НАУКОВА КНИГА
ІНОЗЕМНОЮ МОВОЮ»

КИЇВ
АКАДЕМПЕРІОДИКА
2018

P.I. ANDON, A.Yu. DOROSHENKO, K.A. ZHEREB, O.A. YATSENKO

ALGEBRA - ALGORITHMIC MODELS AND METHODS OF PARALLEL

*PROJECT
«UKRAINIAN SCIENTIFIC BOOK
IN A FOREIGN LANGUAGE»*

KYIV
AKADEMPERIODYKA
2018

Reviewers:

A. V. ANISIMOV, Dean of the Faculty
of Computer Science and Cybernetics of Taras Shevchenko National University of Kyiv,
Corresponding Member of the National Academy of Sciences of Ukraine

I. V. SERGIENKO, Director of the V.M. Glushkov Institute
of Cybernetics of the National Academy of Sciences of Ukraine,
Academician of the National Academy of Sciences of Ukraine

*Approved for publication by the Scientific Council
of the Institute of Software Systems of the NAS of Ukraine
(August, 17, 2017, Protocol No. 9)*

*Publication was funded in the frame of the Target Complex
Program "Creation and Development of Scientific Publishing Complex
of the National Academy of Sciences of Ukraine "*

Andon P. I.

A 54 Algebra-Algorithmic Models and Methods of Parallel Program-
ming; NAS of Ukraine, Institute of Software Systems / P. I. Andon,
A. Yu. Doroshenko, K. A. Zhreb, O. A. Yatsenko. — Kyiv : Akadem-
periodyka, 2018.— 192 p.

ISBN 978-966-360-367-4

The important challenge of modern programming is its mathematization, the develop-
ment of formalized languages for designing algorithms and programs as well as their ab-
stract models. The facilities for design, analysis and implementation of algorithms are es-
pecially actual in connection with essential processes of computerization and automation
of society activities. Such facilities are developed within the framework of algorithmics and
algebraic programming, which are the directions of Ukrainian algebraic-cybernetic school
originating from fundamental works of Academician V. M. Glushkov. The book proposes
formal models, methods and software tools for design of sequential and parallel programs,
which are based on algebras of algorithms and term rewriting paradigm.

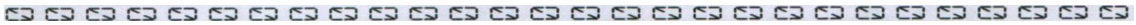
UDC 004.421

ISBN 978-966-360-367-4

©Andon P. I., Doroshenko A. Yu.,
Zhreb K. A., Yatsenko O. A., 2018
© Akademierydyka, design, 2018



CONTENT



LIST OF ABBREVIATIONS AND NOTATIONS.....	5
INTRODUCTION	7

PART 1 ALGEBRAIC MODELS AND METHODS OF DESIGNING ALGORITHMS AND PROGRAMS

CHAPTER 1 . ALGORITHM ALGEBRAS AND PARALLEL PROGRAMMING

1.1. Overview of Algebraic and Parallel Programming Models and Methods.....	11
1.1.1. Algebraic and Insertion Programming	12
1.1.2. Algebra-Algorithmic Programming.....	13
1.1.3. Parallelism and Parallel Computing Models.....	15
1.2. Dijkstra Algebra.....	24
1.3. Algebra of Flowgraphs	29
1.4. Glushkov's Systems of Algorithmic Algebras and Parallel Computing.....	31
1.5. Algebra of Algorithmics.....	37

CHAPTER 2 . FORMALIZED DESIGN OF ALGORITHMS ON THE BASIS OF ALGEBRAS

2.1. Metarules of Schemes Design	40
2.2. Algorithmic Language SAA/1.....	43
2.3. Algebra-Grammatical Models for Generation of Algorithm Schemes.....	46
2.4. Design of Synchronous and Asynchronous Algorithms.....	54

CHAPTER 3 . ALGEBRAIC PROGRAMMING BASED ON REWRITING RULES

3.1. Aspects of Rewriting: Definitions and Examples.....	62
3.1.1. Abstract Rewriting Systems	62
3.1.2. Rewriting Rules.....	66
3.1.3. Rewriting of Fragments.....	67
3.2. Term Rewriting	68
3.2.1. The Alphabet of the System	68
3.2.2. Terms.....	68
3.2.3. Operations on Terms.....	69
3.2.4. Rewriting Rules	71
3.2.5. Rewriting Strategy	72

CONTENT

3.2.6. Interaction with an Environment.....	72
3.2.7. TermWare Language	74
3.3. Existing Software Implementations.....	74
3.3.1. Maude.....	74
3.3.2. Stratego.....	75
3.3.3. ASF + SDF.....	76
3.3.4. TXL.....	77
3.3.5. Jess	77
3.3.6. APS	78
3.3.7. TermWare	78
3.4. Applications of Rewriting Rules Technique for Working with Program Code.....	79
3.4.1. The Main Directions of Use.....	79
3.4.2. The Analysis of a Source Code	79
3.4.3. Refactoring	81
3.4.4. Optimization.....	82
3.4.5. Security	83
3.4.6. Legacy Code.....	83
3.4.7. Domain-Specific Languages.....	84
3.4.8. Extensions of Languages	85
3.4.9. Modelling	85

CHAPTER 4 . ALGEBRA-DYNAMIC MODELS FOR PARALLEL PROGRAM DESIGN

4.1. Discrete Dynamic Systems.....	88
4.2. Sequential Program Execution Model.....	89
4.3. Parallel Program Execution Model	91
4.3.1. GPU Program Model: Block Level.....	91
4.3.2. GPU Program Model: Grid Level.....	92
4.3.3. CPU Program Model.....	92
4.3.4. Applications of Algebra-Dynamic Models.....	93
4.4. Program Transformations Based on Rewriting Rules	94
4.4.1. The Transition from a Sequential Program to GPU Program	94
4.4.2. GPU Program Optimization.....	95
4.4.3. The Transition between High-Level and Low-Level Program Models.....	96

PART 2 THE TOOLS FOR AUTOMATED PROGRAM DESIGN AND THEIR APPLICATIONS

CHAPTER 5 . SOFTWARE TOOLS FOR DESIGN AND SYNTHESIS OF PROGRAMS

5.1. Architecture of the Integrated Toolkit	101
5.2. Dialogue Design of Algorithm Schemes	103

5.3. Generation of Algorithms and Programs	110
5.3.1. Generation of Regular Schemes on the Basis of Hyperschemes.....	111
5.3.2. Synthesis of Programs on the Basis of Algorithm Schemes.....	112
5.4. The Rewriting Rules System	118
5.4.1. The Language of TermWare System	118
5.4.2. The Structure of the Rewriting Rules Toolkit.....	120
5.4.3. Additional Facilities of the Rewriting Rules Toolkit	121
 CHAPTER 6 . AUTOMATED DEVELOPMENT OF EFFICIENT PARALLEL PROGRAMS	
6.1. Design and Parallelization of Programs for Multicore Processors.....	127
6.2. Transformation of Coordination Constructs in Parallel Programs	132
6.3. Development of Parallel Programs for Graphics Processing Units	138
6.3.1. Parallel Summation of Elements of a Numeric Vector.....	138
6.3.2. The Game of Life Problem	142
6.4. Parallelization of Fortran Programs.....	143
6.4.1. Parallelization for Shared-Memory Systems Using OpenMP	145
6.4.2. Performance Evaluation: Test Program and Real-World Example	147
 CHAPTER 7 . PROGRAM DESIGN BASED ON USING ONTOLOGIES AND ALGEBRA OF ALGORITHMS	
7.1. Program Design Ontology.....	151
7.2. Generation of Algorithm Scheme on the Basis of its Ontological Description	155
7.3. Application of the Approach for Designing Meteorological Forecasting Program.....	160
7.3.1. The Mathematical Model of Meteorological Forecasting Problem.....	160
7.3.2. Designing a Parallel Program for Solving the Meteorological Forecasting Problem	161
7.3.3. Design of the Application for Executing the Parallel Program on a Cloud Platform	165
7.3.4. The Results of the Experiment.....	167
 CONCLUSION	169
REFERENCES.....	171
 APPENDIX A. MODELS AND SOURCE CODES OF EXAMPLES	
A.1. Finding the Quantity of Prime Numbers.....	181
A.2. Simulation of One-Dimensional Brownian Motion.....	183
A.3. Parallel Summation of Elements of a Numeric Vector	187