Monographs in Theoretical Computer Science
An EATCS Series

More information about this series at http://www.springer.com/series/776

Kenichi Morita

Theory of Reversible Computing

Springer

Kenichi Morita
Professor Emeritus, Hiroshima University
Hiroshima, Japan

ISSN 1431-2654 ISSN 2193-2069 (electronic)
Monographs in Theoretical Computer Science. An EATCS Series
ISBN 978-4-431-56604-5 ISBN 978-4-431-56606-9 (eBook)
DOI 10.1007/978-4-431-56606-9

Library of Congress Control Number: 2017957823

Printed on acid-free paper

This Springer imprint is published by Springer Nature
The registered company is Springer Japan KK
The registered company address is: Chiyoda First Bldg. East, 8-1 Nishi-Kanda Chiyoda-ku, 101-0065 Tokyo, Japan

To my parents, my wife, my children, and my grandchildren

Preface

A reversible computing system is a "backward deterministic" system such that every state of the system has at most one predecessor. Hence, there is no pair of distinct states that go to the same state. Though its definition is so simple, it is closely related to physical reversibility. The study of reversible computing originated from an investigation of energy dissipation in reversible and irreversible computing systems. Rolf Landauer investigated the relation between reversibility in computing and reversibility in physics in his paper "Irreversibility and heat generation in the computing process" (IBM J. Res. Dev., Vol. 5, pp. 183–191, 1961). He pointed out that an irreversible logical operation inevitably causes energy dissipation in the computing system. Since then, reversible computing has been studied in relation to physical reversibility. Besides the problem of energy dissipation in computing, it is important to know how reversibility can be effectively utilized in computing. This is because future computing devices will surely be implemented directly by physical phenomena in the nano-scale level, and reversibility is one of the fundamental microscopic physical laws of Nature. For this purpose, various models of reversible computing have been proposed and investigated till now.

In this book, reversible computing is studied from the standpoint of the theory of automata and computing. We deal with various reversible computing models belonging to several different levels, which range from a microscopic level to a macroscopic one. They are reversible physical models, reversible logic elements, reversible functional modules composed of logic elements, reversible computing systems such as Turing machines, cellular automata, and others. The purpose of this book is to clarify how computation can be carried out efficiently and elegantly in these reversible computing models. We shall see that even very simple reversible systems have computational universality in spite of the constraint of reversibility. We shall also see various reversible systems in different levels are related each other, i.e., a reversible system in a higher level can be constructed out of those in a lower level. Moreover, the construction methods are often very unique and different from those in the traditional methods. Thus, these computing models as well as the designing methods will give us new insights for future computing systems.

This book is not a comprehensive textbook on reversible computing, but describes mainly the results shown in the papers by myself and my colleagues, which were published between 1989 and 2017. In this book, readers will see how the world of reversible computing works, how reversible computing systems are constructed out of simple reversible primitives, how they are different from the traditional computing systems, and how they can be computationally universal. In fact, we shall see even very simple reversible systems have high capability of computing, and thus reversibility is not a constraint, but a useful property for computing.

This book consists of 14 chapters. Chapter 1 is an introduction. In Chaps. 2–4, reversible logic elements for constructing reversible machines are investigated. In Chaps. 5–8, reversible Turing machines, a standard model of reversible computing systems, are studied. In Chap. 9, some other reversible computing models are investigated. In Chaps. 10–14, reversible cellular automata, a spatiotemporal model of reversible dynamical systems, are studied.

There is no prerequisite knowledge to read this book besides some basics on logic, discrete mathematics and formal languages. But, it is preferable to have some knowledge on the theories of automata and computing. Fortunately, the framework of reversible computing itself is very simple. Therefore, in many cases, readers can easily understand the basic function and the structure of each such system. However, its behavior can be very complex even if the structure of the system is simple. Hence, sometimes, it becomes quite difficult to follow its behavior by using only paper and pencil. In some of these cases, readers can find files in the References that contain computer simulation results of such reversible systems.

More than 50 years have passed since Landauer's paper appeared. Thus, the history of reversible computing is relatively long. But, it is still developing, and there remain many problems to be investigated. Also, even at present, it is not so clear which results will become practically useful in the future. However, the world of reversible computing will lead readers to the new ways of thinking that cannot be found in the traditional design methodologies for computing systems. I hope the theory of reversible computing will stimulate readers' interest, and open new vistas for future computing systems.

Acknowledgments. This work would not have been accomplished without the help and encouragement given by many people in many countries. I would express my highest gratitude to all of them. In particular, I would express my special thanks to Prof. Toshio Mitsui and Prof. Kazuhiro Sugata who guided me in the early days of my research career in Osaka University. Frequent discussions with them led me to the research fields of automata theory, cellular automata, and finally reversible computing. Since then, I had many good colleagues and students in Hiroshima University, Yamagata University, Osaka University, and other places in the world. Working with them was a great pleasure for me. I am grateful to them, especially, Susumu Adachi, Andrew Adamatzky, Artiom Alhazov, Yoichi Fujii, Yoshifumi Gono, Masateru Harao, Takahiro Hori, Katsunobu Imai, Teijiro Isokawa, Chuzo Iwamoto, Atsushi Kanno, Hiroko Kato, Jia Lee, Maurice Margenstern, Genaro J. Martinez, Mitsuya Morimoto, Yuta Mukai, Noritaka Nishihara, Masanori Ogino, Tsuyoshi Ogiro, Ferdinand Peper, Akihiko Shirasaki, Rei Suyama, Keiji Tanaka, Tsuyoshi Tanizawa, Yasuyuki Tojima, Ryoichi Ueno, Satoshi Ueno, Hiroshi Umeo, Yoshikazu Yamaguchi, and Yasunori Yamamoto for their helpful discussions, cooperation, and coauthoring the papers on reversible computing and related fields.

I also express my cordial thanks to the anonymous reviewer for his/her very careful reading and valuable comments. The manuscript was greatly improved by the detailed comments.

This work on reversible computing was supported by JSPS KAKENHI Grant Number JP15K00019.

September 2017 *Kenichi Morita*

Contents

1 Introduction . . . 1
1.1 Reversibility in Physics and Computing . . . 1
1.2 Significance of Reversible Computing . . . 2
1.3 Scope of This Volume . . . 4
1.3.1 Organization of this book . . . 5
1.3.2 Related studies and references . . . 6
1.4 Terminology and Notations . . . 8
References . . . 10

2 Reversible Logic Elements with Memory . . . 15
2.1 Logical Primitives for Reversible Computers . . . 15
2.2 Reversible Logic Element with Memory (RLEM) . . . 16
2.2.1 Rotary element (RE), a typical RLEM . . . 17
2.2.2 Circuit composed of REs . . . 19
2.2.3 Realizing RE in the billiard ball model . . . 22
2.3 Making Reversible Sequential Machines (RSMs) from RE . . . 26
2.3.1 RE-column, a building module for RSMs . . . 26
2.3.2 Composing reversible sequential machines by RE-columns . 27
2.4 Concluding Remarks . . . 30
References . . . 30

3 Classification of Reversible Logic Elements with Memory and Their Universality . . . 31
3.1 Classification of RLEMs . . . 31
3.1.1 Graphical representation of RLEMs . . . 32
3.1.2 Equivalence in RLEMs . . . 35
3.1.3 Degeneracy in RLEMs . . . 40
3.1.4 Classification of 2-, 3- and 4-symbol RLEMs . . . 40
3.2 Universality of All Non-degenerate 2-State RLEMs with Three or More Symbols . . . 41
3.2.1 Realizing RE using non-degenerate 3-symbol RLEMs . . . 41

3.2.2 Making a non-degenerate (*k* 1)-symbol RLEM from a non-degenerate *k*-symbol RLEMs 45
3.3 Systematic Construction of RSMs out of Universal RLEMs 51
3.4 Compact Realization of RSMs Using RLEMs 4-31 and 3-7 53
3.5 Frontier Between Universal and Non-universal RLEMs 56
3.5.1 Definitions on RLEM-circuits 57
3.5.2 Non-universality of three kinds of 2-state 2-symbol RLEMs 60
3.5.3 Universality of combinations of 2-state 2-symbol RLEMs .. 67
3.6 Realizing 4-Symbol RLEMs in the Billiard Ball Model 70
3.7 Concluding Remarks 74
References 74

4 Reversible Logic Gates 77
4.1 Reversible Logic Gates and Circuits 77
4.1.1 Reversible logic gates 78
4.1.2 Reversible combinatorial logic circuits 80
4.1.3 Logical universality of reversible logic gates 82
4.1.4 Clearing garbage information 84
4.1.5 Realization in the billiard ball model 88
4.2 Relation Between Reversible Logic Gates and Reversible Sequential Machines 89
4.2.1 Making Fredkin gate from RE 90
4.2.2 Making RE from Fredkin gate 91
4.2.3 Making reversible sequential machines from Fredkin gate .. 93
4.3 Concluding Remarks 100
References 100

5 Reversible Turing Machines 103
5.1 Turing Machines and Reversibility 103
5.1.1 Basic definitions on reversible Turing machines (RTMs) ... 104
5.1.2 Notion of simulation and computational universality 111
5.1.3 Conversion between the quadruple and quintuple forms 112
5.1.4 Inverse reversible Turing machines 115
5.2 Converting Irreversible Turing Machines to Reversible Ones 119
5.2.1 Three-tape Turing machines 119
5.2.2 Inverse three-tape Turing machines 121
5.2.3 Computational universality of three-tape RTMs 122
5.3 Variations of Reversible Turing Machines 129
5.3.1 Converting RTMs with two-way infinite tapes into RTMs with one-way infinite tapes 129
5.3.2 Converting multi-tape RTMs into one-tape RTMs 134
5.3.3 Converting many-symbol RTMs into two-symbol RTMs ... 138
5.3.4 Converting many-state RTMs into four-state RTMs 143
5.3.5 Converting many-state RTMs into three-state RTMs 148
5.3.6 Computational universality of restricted classes of RTMs ... 154

5.4 Concluding Remarks 155
References 155

6 Making Reversible Turing Machines from Reversible Primitives 157
6.1 Constructing Reversible Turing Machines out of RE 157
6.1.1 Memory cell for two-symbol RTMs 158
6.1.2 Finite control module for two-symbol RTMs 162
6.1.3 RE-circuit that simulates a one-tape two-symbol RTM 164
6.2 Constructing Reversible Turing Machines out of RLEM 4-31 166
6.2.1 Making memory cell out of RLEM 4-31 166
6.2.2 Making finite control module out of RLEM 4-31 168
6.3 Concluding Remarks 171
References 172

7 Universal Reversible Turing Machines 173
7.1 Universal Turing Machines 173
7.2 Tag Systems 176
7.2.1 *m*-tag systems 176
7.2.2 Cyclic tag systems 177
7.3 Small Universal Reversible Turing Machines (URTMs) 181
7.3.1 13-state 7-symbol URTM 182
7.3.2 10-state 8-symbol URTM 185
7.3.3 17-state 5-symbol URTM 187
7.3.4 15-state 6-symbol URTM 189
7.3.5 24-state 4-symbol URTM 191
7.3.6 32-state 3-symbol URTM 193
7.3.7 138-state 2-symbol URTM converted from URTM(24,4) . . . 195
7.3.8 4-state and 3-state URTMs converted from URTM(10,8) and URTM(32,3) 197
7.4 Concluding Remarks 199
References 200

8 Space-Bounded Reversible Turing Machines 203
8.1 Reversibility and Determinism in Space-Bounded Computation 203
8.1.1 Two-tape Turing machine as an acceptor of a language 204
8.1.2 Reversibility and determinism 206
8.1.3 Computation graph 208
8.1.4 Space-bounded TMs 209
8.1.5 Normal forms for TMs 209
8.2 Relation Between Irreversible Deterministic and Reversible Deterministic TMs 213
8.2.1 Halting property of reversible space-bounded TMs 214
8.2.2 Space-efficient reversible simulation of irreversible TMs . . . 215
8.3 Relation Between Reversible Nondeterministic and Reversible Deterministic TMs 222

8.4 Concluding Remarks 226
References 228

9 Other Models of Reversible Machines 229
9.1 Models of Reversible Automata and Machines 229
9.2 Reversible Counter Machines 231
9.2.1 Basic definitions on reversible counter machines 231
9.2.2 Simulating irreversible counter machines by reversible ones 234
9.2.3 Universality of reversible two-counter machine 242
9.3 Reversible Multi-head Finite Automata 249
9.3.1 Basic definitions on two-way multi-head finite automata . . . 249
9.3.2 Converting a multi-head finite automaton into a reversible one with the same number of heads 252
9.4 Concluding Remarks 257
References 259

10 Reversible Cellular Automata 261
10.1 Cellular Automata and Reversibility 261
10.2 Cellular Automata (CAs) 262
10.2.1 Definitions of CAs 263
10.2.2 Examples of CAs 266
10.3 Reversible Cellular Automata (RCAs) 269
10.3.1 Definitions of RCAs 270
10.3.2 Basic properties of RCAs and related CAs 272
10.4 Design Methods for RCAs 275
10.4.1 CAs with block rules 275
10.4.2 Second-order CAs 277
10.4.3 Partitioned CAs (PCAs) and reversible PCAs (RPCAs) 279
10.5 Simulating Irreversible CAs by Reversible CAs 286
10.5.1 Simulating k-dimensional CA by $(k+1)$-dimensional RPCA 286
10.5.2 Simulating one-dimensional CA by one-dimensional RPCA 287
10.6 Making RPCAs from Reversible Logic Elements 293
10.7 Concluding Remarks 295
References 296

11 One-Dimensional Universal Reversible Cellular Automata 299
11.1 Universality in One-Dimensional CAs 299
11.1.1 Ultimately periodic configurations in one-dimensional CAs . 300
11.1.2 Notion of simulation and Turing universality in one-dimensional CAs 302
11.2 One-Dimensional RCAs That Simulate Reversible Turing Machines 303
11.2.1 Simulating RTMs by three-neighbor RPCAs 303
11.2.2 Simulating RTMs by two-neighbor RPCAs 307
11.3 Simple Turing Universal One-Dimensional RPCAs 311

11.3.1 24-state universal RPCA with ultimately periodic configurations . . . 312
11.3.2 98-state universal RPCA with finite configurations . . . 315
11.4 Reversible and Number-Conserving CAs . . . 319
11.4.1 Number-conserving CAs . . . 319
11.4.2 Turing universal reversible and number-conserving CA . . . 322
11.5 Concluding Remarks . . . 327
References . . . 328

12 Two-Dimensional Universal Reversible Cellular Automata . . . 331
12.1 Universality in Two-Dimensional CAs . . . 331
12.1.1 Ultimately periodic configurations in two-dimensional CAs . . . 332
12.1.2 Notion of simulation and Turing universality in two-dimensional CAs . . . 333
12.2 Symmetries in Two-Dimensional PCAs . . . 334
12.3 Simulating Reversible Logic Circuits in Simple RPCAs . . . 335
12.3.1 16-state RPCA model S_1 . . . 335
12.3.2 16-state RPCA model S_2 . . . 337
12.3.3 Turing universality of the two models of 16-state RPCAs . . . 343
12.4 Simulating Reversible Counter Machines in RPCA . . . 343
12.4.1 81-state RPCA model P_3 . . . 344
12.4.2 Basic elements in the RPCA P_3 . . . 345
12.4.3 Constructing reversible counter machine in the RPCA P_3 . . . 350
12.4.4 Turing universality of the RPCA P_3 . . . 361
12.5 Intrinsic Universality of Two-Dimensional RPCAs . . . 361
12.6 Concluding Remarks . . . 364
References . . . 364

13 Reversible Elementary Triangular Partitioned Cellular Automata . . . 367
13.1 Elementary Triangular Partitioned Cellular Automata . . . 367
13.1.1 Triangular partitioned cellular automata (TPCAs) . . . 368
13.1.2 Elementary Triangular Partitioned Cellular Automata (ETPCAs) and reversible ETPCAs (RETPCAs) . . . 374
13.1.3 Dualities in ETPCAs . . . 377
13.2 Conservative RETPCAs and Their Universality . . . 382
13.2.1 Universality of the RETPCA T_{RU} . . . 383
13.2.2 Universality of the RETPCA T_{UR} . . . 394
13.2.3 Universality of the RETPCA T_{RL} . . . 398
13.2.4 Non-universality of the RETPCAs T_{UU}, T_{RR} and T_{LL} . . . 403
13.3 Non-conservative RETPCA T_{0347} That Exhibits Complex Behavior . . . 405
13.3.1 Properties of the RETPCA T_{0347} . . . 406
13.3.2 Glider guns in T_{0347} . . . 414
13.3.3 Universality of the RETPCA T_{0347} . . . 416
13.4 Concluding Remarks . . . 418
References . . . 419

14 Self-reproduction in Reversible Cellular Automata 421
14.1 Self-reproducing Cellular Automata 421
14.2 Self-reproduction in Two- and Three-Dimensional RCAs 424
14.2.1 Two-dimensional model SR_{2D} 424
14.2.2 Three-dimensional model SR_{3D} 444
14.3 Concluding Remarks 447
References 447

Index 449

Acronyms

CA	cellular automaton
CM	counter machine
CTS	cyclic tag system
CTSH	cyclic tag system with halting condition
ECA	elementary cellular automaton
ETPCA	elementary triangular partitioned cellular automaton
GoL	Game of Life
ID	instantaneous description
IDTM	irreversible deterministic Turing machine
INTM	irreversible nondeterministic Turing machine
$\mathscr{L}(\mathscr{A})$	class of languages accepted by the class of automata $\mathscr{A}$
LBA	linear-bounded automaton
MFA	multi-head finite automaton
m-TS	m-tag system
PCA	partitioned cellular automaton
RCA	reversible cellular automaton
RCM	reversible counter machine
RDTM	reversible deterministic Turing machine
RE	rotary element
RETPCA	reversible elementary triangular partitioned cellular automaton
RLEM	reversible logic element with memory
RMFA	reversible multi-head finite automaton
RNTM	reversible nondeterministic Turing machine
RPCA	reversible partitioned cellular automaton
RSM	reversible sequential machine
RTM	reversible Turing machine
SM	sequential machine
TM	Turing machine
TM($S(n)$)	$S(n)$ space-bounded Turing machine
UTM	universal Turing machine
URTM	universal reversible Turing machine