Doctoral thesis

Affine continuous cellular automata and their role in solving density classification problems

Marcin Dembowski

Supervisor: prof. dr. Bernard De Baets, Ghent University, Belgium

Co-supervisor: dr. Barbara Wolnik, Gdansk University, Poland

March 24, 2024

Systems Research Institute
Polish Academy of Sciences
Warsaw, 2024

Declaration
I declare that this thesis was written by me, it does not infringe copyright, it does not violate the legal and material interests of other people.

Oświadczenie (Declaration in Polish)
Ja, poniżej podpisany oświadczam, iż przełożona praca doktorska została wykonana przeze mnie samodzielnie, nie narusza praw autorskich, interesów prawnych i materialnych innych osób.

Preface

I could not finish this thesis without many fantastic people. The journey began with the *Interdisciplinary Doctoral Studies in Mathematical Modeling* at *Gdańsk University* in 2014. I would like to thank Witold Bołt, who encouraged me to try one topic available, namely *Identifiability of cellular automata and related spatio-temporal paradigms*. I just tried that topic as *Cellular Automata* (CAs) was not something new for me. As a software developer and a computer guy, I started with the experiments and simulations and found a few interesting observations and results. Then I started collaboration with a pure mathematical mind – dr. Barbara Wolnik, who helped me to find mathematical tools that could explain those results. Together with prof. dr. Bernard De Baets and dr. Jan M. Baetens from Ghent University in Belgium, we established a strong team where discussion and support helped to proceed with the topic.

Below I present the list of published articles in the chronological order, which are part of this thesis:

[A1] B. Wolnik, M. Dembowski, W. Bołt, J. M. Baetens, and B. De Baets. *The Density Classification Problem in the Context of Continuous Cellular Automata*. In: *Cellular Automata: 12th International Conference on Cellular Automata for Research and Industry, ACRI 2016, Fez, Morocco, September 5-8, 2016. Proceedings*. Ed. by S. El Yacoubi, J. Wąs, and S. Bandini. Cham: Springer International Publishing, 2016, pp. 79–87. ISBN: 978-3-319-44365-2. DOI: 10.1007/978-3-319-44365-2_8

[A2] B. Wolnik, M. Dembowski, W. Bołt, J. M. Baetens, and B. De Baets. *Density-conserving affine continuous cellular automata solving the relaxed density classification problem*. In: *Journal of Physics A: Mathematical and Theoretical* 50.34 (2017), p. 345103. DOI: 10.1088/1751-8121/aa7d86

[A3] M. Dembowski, B. Wolnik, W. T. Bołt, J. Baetens, and B. De Baets. *Affine continuous cellular automata solving the fixed-length density classification problem*. In: *Natural Computing* 17.3 (2018), pp. 467–477. ISSN: 1567-7818. DOI: 10.1007/s11047-017-9631-4

[A4] M. Dembowski, B. Wolnik, W. T. Bołt, J. Baetens, and B. De Baets. *Two-dimensional affine continuous cellular automata solving the relaxed density classification problem*. In: *Journal of Cellular Automata* 14.3-4 (2019), pp. 191–212. ISSN: 1557-5969

[A5] B. Wolnik, M. Dembowski, A. Augustynowicz, and B. De Baets. *A complete description of the dynamics of legal outer-totalistic affine continuous cellular automata*. In: *Nonlinear Dynamics* 110.1 (2022), pp. 589–610. DOI: 10.1007/s11071-022-07642-w

During the preparation of this thesis and papers, the results were also presented at the following international conferences:

[C1] Automata 2015, 21st International Workshop on Cellular Automata and Discrete Complex Systems (Turku, Finland). Bołt W., Dembowski M., Baetens J. M., De Baets B., *Solving the Density Classification Problem by Means of Continuous Cellular Automata*, presented by Witold Bołt.

[C2] ACRI 2016, 12th International conference on Cellular Automata for Research and Industry (Fez, Morocco), Wolnik B., Dembowski M., Bołt W., Baetens J. M., De Baets B. *The density classification problem in the context of continuous cellular automata*, presented by Marcin Dembowski.

[C3] Summer Solstice 2017, 9th International Conference on Discrete Models of Complex Systems (Catania, Italy), Dembowski M., *Solving the fixed-length density classification problem with Affine Continuous Cellular Automata*, presented by Marcin Dembowski.

[C4] Automata 2017, 23rd International Workshop on Cellular Automata and Discrete Complex Systems (Milan, Italy), Dembowski M., Wolnik B., Bołt W., Baetens J. M., and De Baets B., *Solving the relaxed density classification problem by means of two-dimensional Affine Continuous Cellular Automata*, presented by Marcin Dembowski.

[C5] Summer Solstice 2018, 10th International Conference on Discrete Models of Complex Systems (Gdańsk, Poland), Dembowski M., *Solving the relaxed density classification problem by means of two-dimensional Affine Continuous Cellular Automata*, presented by Marcin Dembowski.

[C6] Automata 2018, 24th International Workshop on Cellular Automata and Discrete Complex Systems (Ghent, Belgium), Dembowski M., Wolnik B., Baetens J. M., and De Baets B., *An exploration of the dynamics of one-dimensional outer-totalistic continuous cellular automata*, presented by Marcin Dembowski.

[C7] Automata 2019, 25th International Workshop on Cellular Automata and Discrete Complex Systems (Dresden, Germany), Dembowski M., *An exploration of two-dimensional affine continuous cellular automata rules solving the fixed-size density classification problem*, presented by Marcin Dembowski.

[C8] Summer Solstice 2022 Seminar (Gdańsk, Poland), Augustynowicz A., Wolnik B., Dembowski M., *It happens that a complete description of dynamics of wide class of CAs is possible*, presented by Antoni Augustynowicz.

All the published articles and presentations are part of a teamwork. I authored all visualizations, simulations, experiments in this thesis, published articles and presentations. Also, I analyzed the results collected from experiments that were a base work for further analytical proofs. Proofs of the theorems were achieved thanks to my cooperation with Barbara Wolnik [A1]–[A5] and Antoni Augustynowicz [A5]. The remaining authors of the articles supported the research by discussion, validation of correctness of the simulations or correction of the manuscripts.

This work combines the articles and presentations into one story and extends with additional visualizations and explanations. The chapters, unlike the published articles, explain the topic in an order that first explains the basics and later goes deeper into the details. Chapter 1 provides examples of where and how CAs are used and presents a history of CAs. I wrote it for this thesis and did not publish it in any of the papers. This chapter results from a few readings on the topic in the literature. Chapter 2 gives the formal definitions used in the rest of the work using a unified notation. Those definitions were included in all the published articles and presentations. Chapter 3 extends the definitions for *Affine Continuous Cellular Automata* (ACCAs), a core part of this work. Those definitions were published in [A1]–[A5] and presented at [C2]–[C8]. Chapter 4 explains the focal problem covered by this thesis. The problem was presented in all articles and conferences and formulates the classical and relaxed DCPs that are explained in detail in the subsequent chapters. The next three chapters focus on one-dimensional ACCAs. Chapter 5 contains the latest published results [A5] that were presented at [C6] and [C8] and explains the dynamics of a subset of ACCAs. Chapter 6 focuses on a relaxed version of the DCP and was presented at [C2] and published after in a conference paper [A1]. Extended results for that topic were presented at [C4] and published in [A2]. Chapter 7 covers the classical DCP where the problem was first presented at [C3] and later published in [A3]. The next two chapters extend the topic into two dimensions. Chapter 8 shows results for the relaxed DCP that was first presented at [C4] and at [C5] and published in [A4]. Some experiments and observations from Chapter 9 were presented at [C7] and are not published as an article at this moment. The closing Chapter 10 summarizes and suggests further work that can be done in the area of ACCAs and DCPs and I wrote it specifically for this work.

Any experimental simulations that are presented in this work used various programming languages and techniques. At the beginning of the studies, the *C* language with *Julia* was used. Later, *R* was used for statistics, ending with *Python*. The last language was used with *Cython* and dominated the simulations. The majority of computations were carried out at the Academic Computer Centre in Gdańsk (TASK KDM).

This work was created using LaTeX with Kaobook style. Diagrams were created with Gnuplot as it offered the best customization, resulting in sharp and clean diagrams. The reader can find also a QR-code in a few places. Clicking the code in PDF or scanning it with a mobile phone will direct the reader to a website with examples. Few examples and visualizations presented in this work are available on the GitHub repository under `https://github.com/D3M80L/CA` which contains *Jupyter* notebooks with appropriate examples.

I would like to thank dr. Barbara Wolnik for the excellent support and guidance during the whole research. All the discussions and analytical explanations allowed me to proceed with the work.

Finally, I want to thank my family, especially my wife Margaret and mother Elvira, for supporting me all the time.

Contents

Contents . . . . . **v**

1 Introduction **1**
1.1 Game of Life 1
1.2 Historical overview 2
1.3 Applications 4
1.4 Conclusions 5

2 Preliminaries **7**
2.1 Cellular space 7
2.2 Configurations 8
2.3 Reducing the set $\{0,1\}^\star$ 9
2.4 Neighborhoods 10
2.5 Rules 11
2.6 LUT representation 12
2.7 Rule symmetries 13
2.8 Elementary CAs 15
2.9 Continuous CAs 17
2.10 Conclusions 18

3 Affine Continuous Cellular Automata **19**
3.1 Preliminaries 19
3.2 One-dimensional ACCAs 20
3.3 Outer-totalistic one-dimensional ACCAs 20
3.4 Density-conserving one-dimensional ACCAs 22
3.5 Two-dimensional ACCAs 26
3.6 Density-conserving two-dimensional ACCAs 26
3.6.1 Preliminaries 26
3.6.2 Structure of the space of density-conserving two-dimensional ACCAs 28
3.6.3 Symmetries in the space of density-conserving two-dimensional ACCAs 29
3.6.4 Density-conserving two-dimensional ACCAs with independent y and v 30
3.7 Rotation-symmetric two-dimensional ACCAs 30
3.8 Conclusions 32

4 The Density Classification Problem **33**
4.1 Preliminaries 33
4.2 Classical DCP formulation 34
4.3 Relaxed DCP formulation 36
4.4 Conclusions 40

5 Dynamics of One-Dimensional Outer-Totalistic ACCAs **41**
5.1 Legal outer-totalistic ACCAs 41
5.2 Computer simulations 42
5.3 Dynamical properties of legal outer-totalistic one-dimensional ACCAs 45
5.4 The case $2d - c = 1$: the upper face 47
5.5 The case $2b + c < 1$ and $2d - c < 1$ 49
5.6 The case $2b + c \geq 1$ and $2d - c < 1$ 51
5.6.1 Some technical lemmata 51

5.6.2 The case $(b, c, d) \in \mathcal{P}_0 \setminus (P_1P_3 \cup P_4P_6 \cup P_5P_7)$. 54
5.6.3 The case $(b, c, d) = (0.5, 1, d)$ for $0.5 \leq d < 1$. 57
5.6.4 The case $(b, c, d) = (0.5, 0, d)$ for $0 \leq d < 0.5$. 60
5.6.5 The case $(b, c, d) = (0, 1, d)$ for $0.5 \leq d < 1$. 62
5.7 Conclusions 67

6 Solving the Relaxed One-Dimensional DCP 71
6.1 Dynamical properties of density-conserving ACCAs 71
6.1.1 The case $c = 1$ 71
6.1.2 The case $0 < c < 1$ 72
6.1.3 The case $c = 0$ 76
6.2 Experimental results 77
6.2.1 The case $\rho_0 = \frac{1}{2}$ 78
6.2.2 The case $\rho_0 \in (0, 1)$ 80
6.3 Conclusions 81

7 Solving the Fixed-Length One-Dimensional DCP 83
7.1 Problem formulation 83
7.2 Necessary conditions 86
7.3 Sufficient stopping conditions 91
7.4 Simulation results 93
7.5 Conclusions 96

8 Solving the Relaxed Two-Dimensional DCP 97
8.1 Experimental set-up 97
8.2 Experiments for a 5×5 grid 98
8.3 Experiments for a 23×23 grid 100
8.4 Conclusions 102

9 Solving the Fixed-Length Two-Dimensional DCP 105
9.1 Evolutionary Computation 105
9.2 History 106
9.3 DCP as an optimization problem 107
9.4 Differential Evolution 109
9.4.1 Individuals 109
9.4.2 Population 109
9.4.3 Fitness function 110
9.4.4 Mutation 111
9.4.5 Crossover 111
9.4.6 Selection 112
9.5 Experiments for a 3×3 grid 112
9.5.1 Unique configurations 112
9.5.2 Configurations with density $\rho \approx \frac{1}{2}$ 118
9.5.3 Rotation-symmetric rules 119
9.5.4 Observations 122
9.6 Experiments for a 5×3 grid 123
9.6.1 Rotation-symmetric rules 123
9.6.2 Basin of 3×3 rules 125
9.6.3 Observations on smaller grids 126
9.7 Experiments for a 5×5 grid 127
9.8 Conclusions 128

10 Conclusion **131**
10.1 Summary 131
10.2 Future work 132

Bibliography **135**

Notation **141**

List of Figures

1.1 Conway's Game of Life at different runs. . . . 1
1.2 Citation count of "Cellular Automata" by year from different data sources. . . . 3
1.3 Example of traffic simulation represented by a CA in [23]. . . . 4
1.4 Example simulations of CA models for (a) crowd [33], (b) fluid [34], (c) disease [35] and (d) heart rhythm [36]. . . . 5

2.1 Representation of the one-dimensional cellular space with marked boundaries (dashed line) and the adjacent cells (gray) of cell 4. . . . 7
2.2 Representation of the two-dimensional cellular grid with marked boundaries (dashed line) and the adjacent cells (gray) of cell $(2, 3)$. . . . 8
2.3 Neighborhoods of a given cell i in a one-dimensional cellular space. . . . 10
2.4 The von Neumann neighborhood of a cell (i, j) in a two-dimensional cellular space. . . . 11
2.5 The Moore neighborhood of a cell (i, j) in a two-dimensional cellular space. . . . 11
2.6 Space-time diagrams of rule $\mathbb{f}$:ECA110, starting from configuration $\boldsymbol{x} = 0^3(10)^3 1^5 0^{12} 1^2 0^5$, and its equivalent rules $\mathbb{f}^I$, $\mathbb{f}^C$ and $\mathbb{f}^{IC}$ starting on a mirrored, negated and mirrored-negated configuration $\boldsymbol{x}$, respectively. . . . 15
2.7 Space-time diagrams of selected ECAs for the same initial configuration $\boldsymbol{x} = (0)^3(10)^3(1)^5(0)^{11}$. 16
2.8 Space-time diagram of the one-dimensional diffusion CCA rule $\mathbb{f}_d$ for a configuration $\boldsymbol{x}$ and different cell value palette representations: (a) blue-white gradient and (b) multicolor. . . . 17
2.9 A radial view representation of a configuration $\boldsymbol{x}$ in different iterations t of the one-dimensional diffusion CCA rule $\mathbb{f}_d$. . . . 18

3.1 Parallelogram D that creates a four-dimensional polytope $D \times D$ of the set of outer-totalistic ACCAs. . . . 22
3.2 The pyramid P: parameterization of the set of one-dimensional, density-conserving ACCAs with unit neighborhood radius. . . . 25
3.3 The location of the points $\pi_{\mathbb{f}}$, $\pi_{\mathbb{f}^I}$, $\pi_{\mathbb{f}^C}$ and $\pi_{\mathbb{f}^{IC}}$ corresponding to the local rules $\mathbb{f}$, $\mathbb{f}^I$, $\mathbb{f}^C$ and $\mathbb{f}^{IC}$. 25
3.4 The pyramid P^I: parameterization of the reduced set of one-dimensional, density-conserving ACCAs with unit neighborhood radius. . . . 25
3.5 A scheme of the polytope P with a visualization of five seven-dimensional hyperplanes containing all 65 vertices of P. The vertices that correspond to binary CAs are marked in red, while the others are marked in white. The ACCAs whose local rule is independent of y and v are parameterized by points in the blue triangle. . . . 28
3.6 A scheme of the polytope P with a visualization of five seven-dimensional hyperplanes containing all 65 vertices of P. The vertices that correspond to binary CAs are marked in red, while the others are marked in white. The blue line segment consists of the points corresponding to rotation-symmetric ACCAs. The black point $\pi_{\mathbb{f}_d}$ corresponds to the diffusion ACCA. The ACCAs whose local rule is independent of y and v are parameterized by points in the greyed-out triangle. . . 32

4.1 Flow diagram of the Density Classification Problem. . . . 33

5.1 Parameterization of the set of legal outer-totalistic one-dimensional ACCAs. . . . 41
5.2 Visualization of the dynamics of an ACCA with the local rule parameterized by the point $(b, c, d) = (0.5, 0, 0.25)$ starting from the initial configuration $\boldsymbol{x} = (1)^4(10)^3$. The subsequent radial views show $\mathbb{F}^t(\boldsymbol{x})$ for given t. . . . 45
5.3 Parameterization of the set of double-legal outer-totalistic ACCAs: in the parallelepiped $\mathcal{P}$ (a) in the square $\mathcal{S}$ (b). . . . 47

5.4 Parameterization of the set of legal outer-totalistic one-dimensional ACCAs in the case $2b + c < 1$ and $2d - c < 1$. 50
5.5 The polyhedron $\mathcal{P}_0$. 51
5.6 The graph of the function g in the case $2b + c > 1$. Location of the characteristic points (a) and attractor for an example point $s \in (0, 1)$ (b). 52
5.7 The graph of the function g in the case $2b + c = 1$. The line $y = s$ is tangent to the graph of g at $s_1 = 0$. 53
5.8 The polyhedron $\mathcal{P}_0 \setminus (P_1P_3 \cup P_4P_6 \cup P_5P_7)$. 55
5.9 The edge P_4P_6 in parallelepiped $\mathcal{P}$. 58
5.10 The edge P_1P_3 in parallelepiped $\mathcal{P}$. 60
5.11 The edge P_5P_7 in parallelepiped $\mathcal{P}$. 63

6.1 Parameterization of the set of one-dimensional density-conserving ACCAs with unit neighborhood radius with selected ACCA for the case $c = 1$. 71
6.2 Parameterization of the set of one-dimensional density-conserving ACCAs with unit neighborhood radius with selected ACCAs for the case $0 < c < 1$. 72
6.3 Parameterization of the set of density-conserving ACCAs with unit neighborhood radius with selected ACCAs for the case $c = 0$. 76
6.4 Space-time diagrams of a one-dimensional density-conserving rule parameterized by $(\frac{1}{2}, 0, 0)$ and a configuration with (a) and without (b) substrings of the form $(0)^j$, $j > 1$. 77
6.5 Convergence times obtained for density-conserving ACCAs parameterized by points in the horizontal cross-section of the pyramid P that contains the best solution (black dot) for $n = 25$ cells and cross-section at $c = 0.05$. 78
6.6 Location of the points in the pyramid P corresponding to ACCAs with low convergence time for threshold $\rho_0 = 0.5$ and for configurations of $n = 25$ cells: (a) in the entire pyramid P, and (b) for the diagonal cross-section. 79
6.7 Convergence times obtained for density-conserving ACCAs parameterized by points in the horizontal cross-section of the pyramid P that contains the best solution (black dot) for $n = 149$ cells and cross-section at $c = 0.02$. 79
6.8 Location of the points in the pyramid P corresponding to ACCAs with the lowest convergence time for a given threshold ρ_0, for configurations with $n = 23$ cells: (a) in the entire pyramid P, and (b) for the cross-section at $c = 0.03$. 80
6.9 Minimum convergence time for a given threshold ρ_0 over all ACCAs (black) compared with the convergence time of the diffusion ACCA (red). 81

7.1 Parameterization of the set of double-quiescent local rules with parameters $\eta = 0.1$, $\epsilon = 0.2$, $\gamma = 0.3$. 84
7.2 The solid S (black), *i.e.*, the set of points (η, ϵ, γ) whose coordinates satisfy Eqs. (7.12)–(7.14) versus the set of points (η, ϵ, γ) for which $\gamma \in [-1, 1]$ and $\eta, \epsilon \in [-2, 1]$ (grey). 91
7.3 The location of the points $(\eta, \varepsilon, \gamma)$ listed in Table 7.2 in the solid S. The green point corresponds to Case 5 (as given in Table 7.2). 94
7.4 The location of the points (a, b, c) corresponding to the local rules that belong to DCP*(23) (coloured area) in the polyhedron enclosing all local rules, as defined by the parameters (η, ϵ, γ) listed in Table 7.2. 95
7.5 The location of the points corresponding to $\mathbb{f}_{best}$ and $\mathbb{f}^C_{best}$ in the pyramid for Case 5. 96
7.6 The maximal value (blue) and average value (green) of the cell states and the theoretical convergence limit (red), for the evolution of $\mathbb{f}_{best}$ on the initial configuration $\boldsymbol{x} = 0^9 1^2 (0111)^3$. 96

8.1 The number of configurations $\boldsymbol{x} \in \mathcal{I}_C$ with a given time difference $\Delta T(\boldsymbol{x})$ between rule $\mathbb{f}_d$ and $\mathbb{f}_\star$ for a 5×5 grid. 99
8.2 Configurations $\boldsymbol{x}$ for given values of $\Delta T(\boldsymbol{x})$ and a 5×5 grid. 99

8.3 Results for a 5×5 grid and rotation-symmetric ACCAs. 100
8.4 Results for density-conserving ACCAs whose local rule is independent of y and v for a 5×5 grid . 100
8.5 The number of configurations $\boldsymbol{x} \in \mathcal{I}_C$ with a given difference of convergence time $\Delta T(\boldsymbol{x})$ between rule $\mathbb{f}_d$ and $\mathbb{f}_\star$ for a 23×23 grid. 101
8.6 Configurations $\boldsymbol{x}$ for given values of $\Delta T(\boldsymbol{x})$ for a 23×23 grid. 101
8.7 Results for rotation-symmetric density-conserving ACCA for a 23×23 grid. 102
8.8 Results for density-conserving ACCA whose local rule is independent of y and v for a 23×23 grid. 102

9.1 Diagram of an evolution process in evolutionary computation. 105
9.2 Graphical representation of a population $\mathcal{P}$ of individuals $\iota_0, \ldots, \iota_{p-1}$. 109
9.3 Graphical representation of a differential mutation. 111
9.4 Distribution of individual fitness for 1000 DE runs for the problem of finding an ACCA that solve the DCP using all binary configurations in the training set on the 3×3 grid. 113
9.5 Fitness value of the best individual in a generation with marked fitness of the best individual in the validation process for an individual representing an ACCA solving the DCP on the 3×3 grid. 113
9.6 Fitness histograms of individuals from the population in selected generations of the DE run for the problem of finding an ACCA that solves the classical DCP for the 3×3 grid. 114
9.7 Fitness value of the best individual in a generation with marked fitness of the best individual in the validation process for an individual representing an ACCA solving the DCP on the 3×3 grid. 115
9.8 LUT coefficient distribution in selected generations for a DE run for the problem of finding an ACCA that solves the classical DCP for the 3×3 grid. Blue rotated squares represent the coefficient values of the best individual in the generation. 116
9.9 LUT coefficient distribution of the 959 best ACCAs found in DE experiments for the problem of finding an ACCA that solves the classical DCP for the 3×3 grid. 116
9.10 Fitness value of the best individual in a generation with marked fitness of the best individual in the validation process for an individual representing an ACCA that does not completely solve the classical DCP on the 3×3 grid. 117
9.11 Configuration $\boldsymbol{x}$ and a sample evolution of an ACCA rule found by the DE algorithm that does not solve the classical DCP on the 3×3 grid. 117
9.12 Distribution of individual fitness for 1000 DE runs for the problem of finding an ACCA rule that solves the DCP using only binary configurations with a density close to $\frac{1}{2}$ in the training set on the 3×3 grid. 118
9.13 Fitness value of the best individual in a generation with marked fitness of the best individual in the validation process for an individual representing an ACCA rule solving the classical DCP on the 3×3 grid with only configurations with the density close to $\frac{1}{2}$ used during evolution. . . . 119
9.14 Fitness value of the best individual in a generation solving the classical DCP on the 3×3 grid. Only configurations with the density close to $\frac{1}{2}$ were used in the learning set. 119
9.15 Configuration $\boldsymbol{x}$ and sample iterations of an ACCA rule found by the DE algorithm using only configurations with a density close to $\rho \approx \frac{1}{2}$ for the 3×3 grid that does not evolve to the correct homogeneous configuration. 120
9.16 Distribution of individual fitness for 1000 DE runs for the problem of finding an RS ACCA rule that solves the DCP using only binary configurations with a density close to $\frac{1}{2}$ in the training set on the 3×3 grid. 120
9.17 Fitness value of the best individual in a generation of an ACCA solving the DCP on the 3×3 grid where configurations with the density near $\frac{1}{2}$ and only individuals representing rotation-symmetric rules were used during evolution. 121
9.18 Two configurations and sample iterations of an RS ACCA rule found by the DE algorithm for the 3×3 grid that does not evolve to the correct homogeneous configuration. 121
9.19 Distribution of the LUT coefficients of 962 RS ACCA rules, found in the DE experiments, that solve the classical DCP on the 3×3 grid. 122

9.20 Example of configurations whose time t_ϵ to reach the threshold value ϵ is much better (shorter) compared to the time t_H where the evolution reaches a homogeneous configuration of 0s or 1s. 122
9.21 Distribution of generations in relation to the population size for the DE algorithm and a problem of finding an ACCA that solves a classical DCP for the 5×3 grid with termination condition $T = 20$. 124
9.22 LUT coefficient distribution of the 996 best RS ACCAs found in DE experiments for the problem of finding an ACCA that solves a classical DCP for the 5×3 grid. 124
9.23 Fitness value of the best individual in a given generation of an ACCA that does not fully solve the classical DCP on the 5×3 grid. 125
9.24 Configurations that were not classified by the fittest RS ACCA rule found by the DE algorithm for the classical DCP on the 5×3 grid. 125
9.25 Distribution of DE generations in relation to the population size for the problem of finding an ACCA rule that solves the classical DCP for the 5×3 grid with termination condition $T = 20$ where the initial population consists of randomly chosen RS ACCA that solve the classical DCP for the 3×3 grid. 126
9.26 Examples of 3×3 configurations that were not classified by a RS ACCA rule that solves all configurations on the 5×3 grid. 126
9.27 Distribution of generations for population size $|\mathcal{P}| = 40$ and termination condition $T \in \{30, 35\}$ for the DE algorithm and the problem of finding an ACCA that solves the classical DCP for the 5×5. 127
9.28 LUT coefficient distribution of the 2860 best RS ACCAs found in DE experiments for the problem of finding an ACCA that solves the classical DCP for the 5×5 grid. 128

List of Tables

2.1 General form of the LUT representation of the local rule with $Q = \{0, 1\}$, unit neighborhood radius ($r = 1$) and one-dimensional cellular space. 12
2.2 General form of the LUT of the local rule of a two-dimensional CA with the von Neumann neighborhood and $Q = \{0, 1\}$. 13
2.3 Selected ECA rules with their LUT representation and Wolfram class. 15

3.1 Expression of the parameters $a^T, \ldots, h^T$ corresponding to the two-dimensional local rule $\mathbb{f}^T$ for the transformations I, R and C. 29
3.2 The labeling of the vertices of the polytope P. The transformations considered are: Id – identity, H, I – horizontal and vertical axial flips along perpendicular bisectors ($H = IR^2$), A_1, A_2 – axial flips along diagonal bisectors ($A_1 = RI, A_2 = IR$), R, R^2, R^3 - rotation by 90, 180 and 270 degrees), C – conjugation. 30
3.3 LUT of a two-dimensional, density-conserving and rotation-symmetric ACCA rule with the von Neumann neighborhood parameterized by $\alpha \in [0, \frac{1}{4}]$. 32

5.1 Analytical expressions of the local rules corresponding to the extreme points in the set of legal outer-totalistic one-dimensional ACCAs. 42
5.2 Examples of points (b, c, d) representing five distinct dynamics observed in the simulation of double-legal outer-totalistic ACCAs lying on the upper face $P'_2P'_3P'_6P'_7$ of the parallelepiped $\mathcal{P}$. . 48
5.3 Classes of ACCA rules with distinct dynamical behavior. 68

6.1 The LUT of a one-dimensional density-conserving ACCA parameterized by $(0, 0, c)$. 75
6.2 The LUT of a one-dimensional density-conserving ACCA parameterized by $(a, 0, 0)$. 77

7.1 The number of neighborhoods N_1, N_2, N_3 and N_7 in the configurations considered in the proof of Lemma 7.1.1. 87
7.2 Parameters η, ϵ, γ of the local rules that solve the DCP* for $N = 23$. 94
7.3 The most performant rules for $N = 23$. 94

8.1 Parameters of the best-performing rules found for a 5×5 grid. 98
8.2 Parameters of the best-performing rules found for a 23×23 grid. 101

9.1 LUT of a two-dimensional ACCA rule that solves the classical DCP for the 3×3 grid found by the DE algorithm. 113
9.2 An example of a LUT of a two-dimensional ACCA rule found by the DE algorithm that do not solve the classical DCP for all configurations on the 3×3 grid. 117
9.3 LUT of a two-dimensional ACCA rule that solves the classical DCP for the 3×3 grid found by the DE algorithm where only configurations with a density close to $\frac{1}{2}$ were used during the evolution. 118
9.4 An example of a LUT of a two-dimensional ACCA rule found by DE algorithm that does not solve the classical DCP for all configurations on the 3×3 grid, but solves all configurations with the density $\rho \approx \frac{1}{2}$. 119
9.5 A LUT of a two-dimensional rotation-symmetric ACCA rule found by the DE algorithm in 40-th generation that solves the classical DCP for all configurations on the 3×3 grid. 121
9.6 A LUT of a two-dimensional rotation-symmetric ACCA rule found by the DE algorithm in the 40-th generation that does not solve the classical DCP for all configurations on the 3×3 grid. . 121
9.7 Relation between population size $\mathcal{P}$ and termination condition for a number of configuration iterations T and the number of failed experiments in 1000 runs of experiments for problem of finding RS ACCA rules that solve the classical DCP on the 3×3 grid. 123
9.8 Relation between population size $\mathcal{P}$ and a termination condition for a number of configuration iterations T and the number of failed experiments in 1000 runs of experiments for the problem of finding RS ACCA rules that solve the classical DCP on the 5×3 grid. 124
9.9 The best LUT of a two-dimensional rotation-symmetric ACCA rule found by DE algorithm that does not solve the classical DCP for all configurations on the 5×3 grid. 125
9.10 The LUT of a two-dimensional rotation-symmetric ACCA rule found by DE algorithm that solves the classical DCP for all configurations on the 5×3 grid. 127

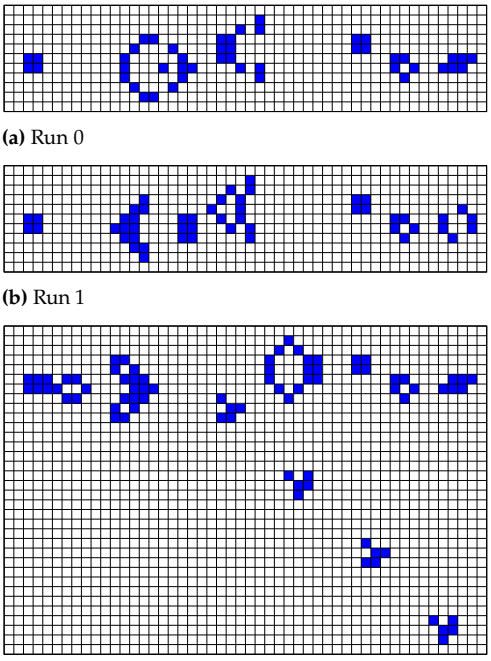
Introduction 1

1.1 Game of Life 1
1.2 Historical overview 2
1.3 Applications 4
1.4 Conclusions 5

In recent years, the entire world focused its attention on the COVID-19 pandemic. The first case of COVID-19, a disease caused by a virus, was recorded in China, from whence the virus spread worldwide within a few weeks, leading to a pandemic. It is not the first time that a disease poses a threat to the human population. The Spanish flu killed millions of people in 1918. The well-known Bubonic plague, also referred to as the Black Death, caused the death of millions of people in the Middle Ages. There are a few factors that influence the speed of the virus spread: density of the population, restrictions, vaccinations. All the pandemics started from a certain point and propagated via neighbors to the rest of the population. Based on the factors listed above, it is possible to create a simple Cellular Automaton (CA) model that will represent any virus spread.

1.1 Game of Life

A very simple and well-known example of a CA model is Conway's *Game of Life*.



(a) Run 0

(b) Run 1

(c) Run 168

Figure 1.1: Conway's Game of Life at different runs.

In this model, a two-dimensional grid storing binary values, representing dead or living cells, is initialized. An exemplary representation of the results at different runs for a *Gosper glider gun* pattern is presented in Figure 1.1. The initial pattern, shown in Figure 1.1a, contains a few objects that can generate other objects at certain runs. In the next runs, we can see

unchanging objects (like blocks in the left part of the board), oscillators (in the right part of the board) and glider objects (the bottom part in Figure 1.1c). To see how fragile the game is, we can try to disable one cell or add noise in some places. The game runs will be completely different, with all objects being destroyed as a result. This CA created by John Conway[1] was popularized in [6] where the game rules were described:

> *First note that each cell of the checkerboard (assumed to be an infinite plane) has eight neighboring cells, four adjacent orthogonally, four adjacent diagonally. The rules are:*
>
> - *Survivals. Every counter with two or three neighboring counters survives for the next generation.*
> - *Deaths. Each counter with four or more neighbors dies (is removed) from overpopulation. Every counter with one neighbor or none dies from isolation.*
> - *Births. Each empty cell adjacent to exactly three neighbors–no more, no fewer–is a birth cell. A counter is placed on it at the next move.*
>
> *It is important to understand that all births and deaths occur simultaneously. Together they constitute a single generation or, as we shall call it, a "move" in the complete "life history" of the initial configuration. ([6, Martin Gardner])*

1: John Conway (26 December 1937 – 11 April 2020). He was an English mathematician active in many areas of research like game, group, number theory.

As not only two-dimensional CA models are available, we can extend the definition to any dimension. There are a few things in common not only between the game and virus spread, but also across all CAs in general. In both cases local interactions have influence on the global state, each change is defined by a rule and even a small change can give completely different results at the end.

1.2 Historical overview

The end of the 1940s can be indicated as the very beginning of CAs. Stanislaw Ulam[2] studied the growth of crystals at the Los Alamos National Laboratory in the USA using a lattice network model. At the same time and place, John von Neumann[3] was working on the problem of self-replicating systems, a universal constructor. Based on Ulam's suggestions, von Neumann created a two-dimensional, 29-state model where time and space were discrete [7]. The original concept of CA is credited to Ulam, while the development and expansion is credited to von Neumann [8]. Nowadays we can see that the idea of self-replicating machines is similar to the replication of DNA.

2: Stanislaw Ulam, (3 April 1909 – 13 May 1984) was a Polish-American mathematician. He not only suggested the concept of CAs but also invented the Monte Carlo computation algorithm.

3: John von Neumann (28 December 1903 – 8 February 1957) was a Hungarian-American mathematician, engineer, and computer scientist. He developed the game theory.

In 1969, Konrad Zuse[4] suggested that the universe is directed by discrete rules and controlled by a kind of CAs [9]. In 1970, John Conway was fascinated by von Neumann's work in the 1940s. He used a simpler model, with only two states and shared his idea with Martin Gardner[5]. This idea was published in Scientific American in the Mathematical Games column [6] as *the Conway's idea of "Game of life"*, whose rules were presented in Section 1.1.

4: Konrad Zuse (22 June 1910 – 18 December 1995) was a German computer scientist, founder of Zuse KG (first computer company in Germany). His biggest invention was the world's first programmable, Turing-Complete computer Z3 in 1941.

5: Martin Gardner (21 October 1914 – 22 May 2010) was an American popular mathematics and science writer. He was best known for popularizing mathematics through the Mathematical Games column in *Scientific American*, a popular science magazine.

> *The game made Conway instantly famous, but it also opened up a whole new field of mathematical research, the field of cellular automata. (Martin Gardner)*

From that day on, more computer scientists were committed not only to the game and finding interesting objects, like the Glider gun, but also to the overall CA topic. In the sixties and seventies, researchers were more focused on visual analysis of the patterns that were produced by CAs.

In 1978, a model of excitable media was developed by Greenberg and Hastings [10]. It is a two-dimensional CA, which has cells with only three states and four neighbors in the simplest case. Each cell is assigned to one of the states: "resting", "excited" or "refractory".

In 1980, Stephen Wolfram[6] investigated and finally classified a class of elementary cellular automata. He suggested that one elementary rule may be universal and formulated the concept of intrinsic randomness. This fact was proved by Matthew Cook[7] who showed that Rule 110 is Turing-complete [11][8]. From 1992 to 2002, Wolfram worked on a detailed study of CAs, the results of which were published in the book "A new kind of Science" [12]. In his monograph, Wolfram related CAs to all disciplines of science, but he was also criticized for the lack of rigorous mathematical tests.

> *In ANKS [A New Kind of Science] Wolfram says that "... the core of this book can be viewed as introducing a major generalization of mathematics" [...]. In this he is entirely mistaken, but there are at least two ways in which he has benefited mathematics: he has helped to popularize a relatively little-known mathematical area (CA theory), and he has unwittingly provided several highly instructive examples of the pitfalls of trying to dispense with mathematical rigor. ([13, Lawrence Gray])*

Finally, in 2014 Gerard 't Hooft[9] presented a CA interpretation of quantum mechanics [14].

The number of research papers, authors and topics in the theory of CAs grows over time. Figure 1.2 shows the results from Wolfram's research [15]

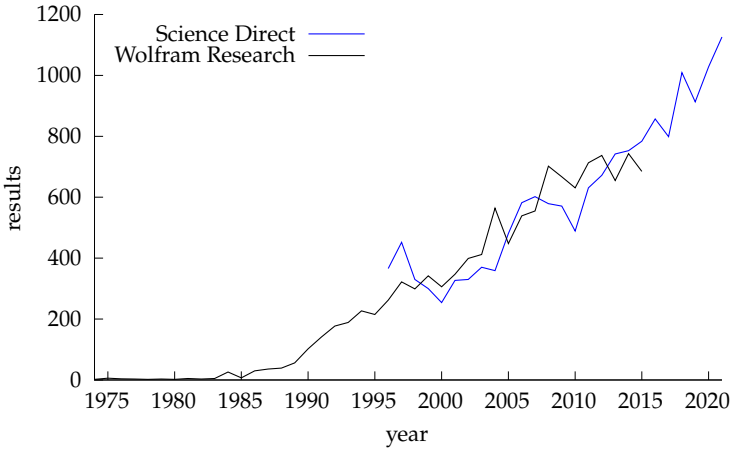


Figure 1.2: Citation count of "Cellular Automata" by year from different data sources.

and search results from the Science Direct portal. It is difficult to recall all historical notes, discoveries and scientific work related to CAs, therefore we will limit ourselves to only a few examples in the next section.

6: Stephen Wolfram (29 August 1959) is a British-American computer scientist and CEO of the Wolfram Research software company. One of the famous products of Wolfram Research is the technical computing system Mathematica, and the answering engine Wolfram Alpha.

7: Matthew Cook (February 7, 1970) is an American mathematician and computer scientist. In 1990 he worked as a research assistant to Stephen Wolfram in the *Wolfram Research* laboratories and assisted with the work on Wolfram's book *A New Kind of Science*.

8: The proof was presented at the Santa Fe Institute conference before the publication of Wolfram's book. This caused a violation of the NDA between him and *Wolfram Research* and resulted in the blocking of the conference proceedings publication. Finally, the proof was published in 2004, two years after *A New Kind of Science* was published.

9: Gerard 't Hooft (5 July 1946), a Dutch theoretical physicist. In 1999 he shared the Nobel Prize in Physics.

1.3 Applications

Because of their simplicity, CAs are suitable for hardware implementations and have attracted substantial interest as models of complex phenomena in different areas of research. Biology, hydrology, geology, agriculture, material sciences, pedestrian dynamics, transport are just a few examples where CAs are used. Let us have a brief look at some examples from the above disciplines.

The chaotic properties of Rule 30 (which we will also mention later) have been used as a pseudo-random number generator [16]. Such properties motivated researchers to explore and study cryptographic properties of CAs [17–21].

Traffic simulation is another example where CAs are used [22–24]. A visual representation available in [23] is presented in Figure 1.3. For

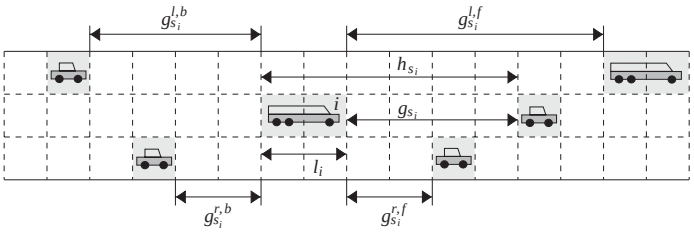


Figure 1.3: Example of traffic simulation represented by a CA in [23].

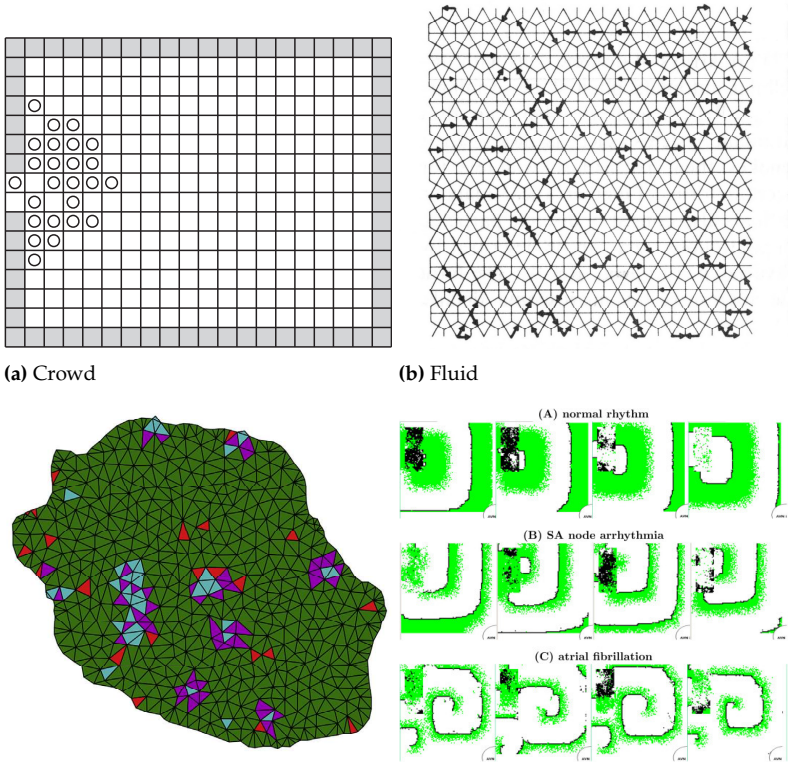
example, in [24]

> *[the] model is defined on a one-dimensional array of L sites and with open or periodic boundary conditions. Each site may either be occupied by one vehicle, or it may be empty. Each vehicle has an integer velocity [. . .]. For an arbitrary configuration, one update of the system consists of the following four consecutive steps [. . .]:*
>
> - *Acceleration [. . .].*
> - *Slowing down [. . .].*
> - *Randomization [. . .].*
> - *Car motion [. . .].*
>
> *([24, Kai Nagel and Michael Schreckenberg])*

For pedestrian simulations a CA is used as a model to optimize evacuation from certain areas [25–32]. A snapshot of a simulation available in [33] is presented in Figure 1.4a. Fluid and wave propagation are other examples [34, 37, 38] where CAs are used. For example, a CA fluid model on a hexagonal grid proposed in [34] is presented in Figure 1.4b.

Fire spread [39–41] or disease spread [35, 42], as presented in Figure 1.4c, can also be modelled using a CA model [43, 44].

Heart research is the last but not least example, where CAs can be used [36, 44]. For example, in Figure 1.4d from [36] the heart rhythm in different CA simulations is presented.



(a) Crowd **(b)** Fluid

(c) Disease **(d)** Heart

Figure 1.4: Example simulations of CA models for (a) crowd [33], (b) fluid [34], (c) disease [35] and (d) heart rhythm [36].

1.4 Conclusions

We can see that the multidisciplinary use of CA models stems from the fact that CAs reflect all global properties of a given phenomenon as resulting from local interactions. In the following chapters we will give a more formal definition of a CA and focus on one of the most famous problems in the theory of CAs.

Preliminaries 2

2.1 Cellular space 7
2.2 Configurations 8
2.3 Reducing the set $\{0,1\}^\star$. 9
2.4 Neighborhoods 10
2.5 Rules 11
2.6 LUT representation 12
2.7 Rule symmetries 13
2.8 Elementary CAs 15
2.9 Continuous CAs 17
2.10 Conclusions 18

In this chapter, we give formal definitions of the notations used in this work. The items are indexed from zero, as this simplifies the work with arrays and reduces issues related to the implementation in any modern programming language. This approach also simplifies modular arithmetic.

2.1 Cellular space

We consider finite CAs with periodic boundary conditions on a rectangular grid only.

Let n be a natural number. We define the one-dimensional cellular space as a grid C_1 with periodic boundary conditions

$$C_1 = (\mathbb{Z}/n\mathbb{Z}) = \{0, 1, \ldots, n-1\} .$$

Hence, we assume that the cells are arranged in a circular linear array. The cardinality of C_1 is given by $|C_1| = n$. We refer to a cell as a single i where $i \in \mathbb{Z}/n\mathbb{Z}$. Each cell in C_1 has exactly two adjacent cells, namely $i-1$ and $i+1$. A graphical representation of this cellular space is given in Figure 2.1.



Figure 2.1: Representation of the one-dimensional cellular space with marked boundaries (dashed line) and the adjacent cells (gray) of cell 4.

For a two-dimensional CA, let n, m be natural numbers greater than two. We define the two-dimensional cellular space as a two-dimensional grid C_2 with periodic boundary conditions

$$C_2 = (\mathbb{Z}/n\mathbb{Z}) \times (\mathbb{Z}/m\mathbb{Z}) = \{0, 1, \ldots, n-1\} \times \{0, 1, \ldots, m-1\} .$$

Hence, the cardinality of C_2 is given by $|C_2| = n\,m$. Furthermore, we refer to a cell as a pair (i, j), where $i \in \mathbb{Z}/n\mathbb{Z}$ and $j \in \mathbb{Z}/m\mathbb{Z}$. As a consequence of the imposed periodic boundary conditions, each cell in C_2 has exactly four adjacent cells, namely $(i-1, j)$, $(i, j-1)$, $(i+1, j)$ and $(i, j+1)$. A graphical representation of this two-dimensional cellular space is given in Figure 2.2.

Depending on the context, we will denote the one- or two-dimensional cellular space by C.

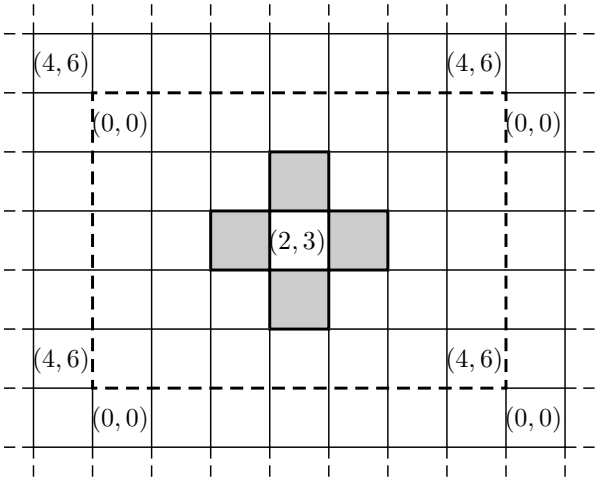


Figure 2.2: Representation of the two-dimensional cellular grid with marked boundaries (dashed line) and the adjacent cells (gray) of cell (2, 3).

2.2 Configurations

We define a set $Q \subseteq \mathbb{R}$ (containing zero) that specifies the values a cell can have. We focus on the discrete, binary state set $Q_B = \{0, 1\}$ and the real-valued state set $Q_{\mathbb{R}} = [0, 1]$. A *configuration* is any mapping from C to Q. The set of all configurations for a given C and Q is denoted by $\mathcal{X}$, *i.e.*, $\mathcal{X} = Q^C$. For a configuration $\boldsymbol{x} \in \mathcal{X}$ in the one-dimensional case, the state of cell i is denoted by x_i, and by $x_{i,j}$ respectively for a cell (i, j) from the two-dimensional cellular space C_2.

In the case of the one-dimensional cellular space C_1, the set $\mathcal{X}$ is identified with Q^n, where n is the number of cells in C_1. Elements of this set will be referred to as configurations, strings or vectors, depending on the context. Additionally, we introduce the following two sets of configurations:

$$[0, 1]^* = \bigcup_{n=1}^{\infty} Q_{\mathbb{R}}^n \,, \quad \{0, 1\}^* = \bigcup_{n=1}^{\infty} Q_B^n \,.$$

The first set consists of all possible configurations, while the second one only comprises the binary configurations. Of course $\{0, 1\}^\star \subset [0, 1]^\star$.

We prefer to write the elements of $[0, 1]^\star$ without brackets or separators. For example, instead of $(0, 1, 0, 0, 0, 1, 1)$, we will just write 0100011. Moreover, for the sake of convenience, for $\boldsymbol{x}, \boldsymbol{y} \in \mathcal{X}$, we will use $\boldsymbol{xy}$ to refer to the concatenation of $\boldsymbol{x}$ and $\boldsymbol{y}$, *i.e.*, if $\boldsymbol{x} = (x_0, \ldots, x_{n-1})$ and $\boldsymbol{y} = (y_0, \ldots, y_{k-1})$, then $\boldsymbol{xy} = (x_0, \ldots, x_{n-1}, y_0, \ldots, y_{k-1})$. In addition, for $j \in \mathbb{N}$, the configuration $(\boldsymbol{x})^j$ is defined as a concatenation of j instances of $\boldsymbol{x}$, *i.e.*:

$$(\boldsymbol{x})^j = \underbrace{\boldsymbol{x}\boldsymbol{x}\cdots\boldsymbol{x}}_{j \text{ times}} \,.$$

This notation will also be used in the context of individual symbols $s \in Q$ to denote repetition, *i.e.*, $(s)^j = (s, \ldots, s) \in Q^j$.

If $\boldsymbol{x} \in Q^n$, then by $|\boldsymbol{x}|$ we denote the length of the configuration, *i.e.*, $|\boldsymbol{x}| = n$. We also define two functions, the *sum* of the states of configuration

$$\sigma(\boldsymbol{x}) = \sum_{i \in C_1} x_i \,, \tag{2.1}$$

and the *density* of a configuration

$$\rho(\boldsymbol{x}) = \frac{\sigma(\boldsymbol{x})}{|\boldsymbol{x}|} \,. \tag{2.2}$$

Clearly, for $\boldsymbol{x} \in \{0,1\}^\star$, the density $\rho(\boldsymbol{x})$ is nothing else than the fraction of 1s in $\boldsymbol{x}$.

In the case of the two-dimensional cellular space C_2, the set $\mathcal{X}$ is identified with $\mathcal{Q}^{nm}$. Similarly as in the one-dimensional case, we introduce the following two sets, containing all possible configurations on all possible two-dimensional grids:

$$[0,1]^\star = \bigcup_{n=1}^{\infty}\bigcup_{m=1}^{\infty} \mathcal{Q}_{\mathbb{R}}^{nm} \,, \quad \{0,1\}^\star = \bigcup_{n=1}^{\infty}\bigcup_{m=1}^{\infty} \mathcal{Q}_{B}^{nm} \,.$$

Although we use the same symbols as in the one-dimensional case, this should not lead to confusion, because it will be clear from the context which set are referring to. If $\boldsymbol{x} \in \mathcal{Q}^{nm}$, then $|\boldsymbol{x}|$ denotes the size of the configuration, *i.e.*, $|\boldsymbol{x}| = nm$. The sum of a configuration $\boldsymbol{x}$ is defined as

$$\sigma(\boldsymbol{x}) = \sum_{(i,j)\in C_2} x_{i,j} \,.$$

2.3 Reducing the set $\{0,1\}^\star$

When using binary configurations on a space C with periodic boundary conditions, we can observe an interesting fact. Consider the binary one-dimensional configuration $\boldsymbol{x} = 01011000$. Repeatedly concatenating the configuration by adding the same representation to the left or the right gives us a configuration $\boldsymbol{y} = (\boldsymbol{x})^\infty$. The same configuration $\boldsymbol{y}$ can be obtained when a different configuration is used, for instance $\boldsymbol{z} = 00001011$ or any other shift of the configuration $\boldsymbol{x}$ to the left or the right. This observation brings improvements in many experiments based on configurations, as we can reduce the number of computations on the set of configurations by considering only a single representative. In further experiments, we will refer to $\mathcal{I}_C$ as a set of all initial, binary configurations with representatives only. The representative configuration is the one that has the lowest decimal representation found when shifting the configuration. This means that any binary configuration $\boldsymbol{x} \in \mathcal{Q}_B$ has a unique integer representation,

$$\mathrm{int}(\boldsymbol{x}) = \sum_{i=0}^{|\boldsymbol{x}|-1} 2^{|\boldsymbol{x}|-i-1} x_i \,.$$

To illustrate the approach, let us explain it on an example.

Example 2.3.1 Let us consider the binary configuration $\boldsymbol{x}_0 = 110$. Shifting this configuration to the right, keeping the boundary conditions, gives us two other configurations $\boldsymbol{x}_1 = 011$ and $\boldsymbol{x}_2 = 101$. Our goal is to pick the representative. Computing the integer representation gives us the answer that $\boldsymbol{x}_1$ is the representative, as it has the smallest integer number assigned, because $\mathrm{int}(\boldsymbol{x}_1) = 3 < \mathrm{int}(\boldsymbol{x}_0) = 5 < \mathrm{int}(\boldsymbol{x}_2) = 6$.

We can approximate the size of $\mathcal{I}_C$ as $\approx \frac{2^{|x|}}{|x|}$, and get the exact size by using Burnside's Lemma and Polya's enumeration theorem. It is also possible to enumerate the representatives using a naive algorithm, as presented in Example 2.3.1. We need to recall that enumerating all configurations is becoming impossible as the set still grows exponentially with the length of the configuration x. In that case, for long configurations, we can randomly choose a configuration x and shift it to a representative, as the probability of selecting randomly two configurations of the same representative is lowering with the length of the configuration. In further chapters, we will also present other ideas on how to reduce the set $\mathcal{I}_C$.

2.4 Neighborhoods

We will only focus on symmetric neighborhoods. As a result of the imposition of periodic boundary conditions, every cell has $2r$ neighbors, where the number r denotes the radius of the neighborhood, *i.e.*, r left and r right neighbors for the one-dimensional cellular space, and additionally r top and r bottom neighbors for the two-dimensional cellular space. In this setting, $\mathcal{R} = 2^d r + 1$ is the size of the neighborhood including the cell, where d is the dimension of the cellular space. Different neighborhoods in the one-dimensional case are presented in Figure 2.3.



(a) $r = 1, \mathcal{R} = 3$

(b) $r = 2, \mathcal{R} = 5$

Figure 2.3: Neighborhoods of a given cell i in a one-dimensional cellular space.

Unless otherwise stated, for the one-dimensional cellular space, the neighborhood consists of the cell itself, and its left and right neighbor, *i.e.*, the radius of the neighborhood is $r = 1$ and $\mathcal{R} = 3$. The states of the cells in the neighborhood of the cell i are represented as x_{i-1}, x_i, x_{i+1}. This neighborhood is presented in Figure 2.3a. In this case, we define $\mathcal{D}_1$ as the domain of this neighborhood

$$\mathcal{D}_1 = \{x\, y\, z \,|x, y, z \in Q\} \ .$$

For the two-dimensional cellular space, the neighborhood, like in the one-dimensional case, comprises the cell itself, its left and right neighbors, but also its top and bottom neighbors, *i.e.*, the radius of the neighborhood still equals $r = 1$, but $\mathcal{R} = 5$. This neighborhood is called the *von Neumann neighborhood*, as it was used by John von Neumann in his works, and is presented in Figure 2.4. The states of the cell (i, j) in this neighborhood are represented as

$$\begin{array}{ccc} & x_{i,j-1} & \\ x_{i-1,j} & x_{i,j} & x_{i+1,j} \\ & x_{i,j+1} & \end{array} \ .$$

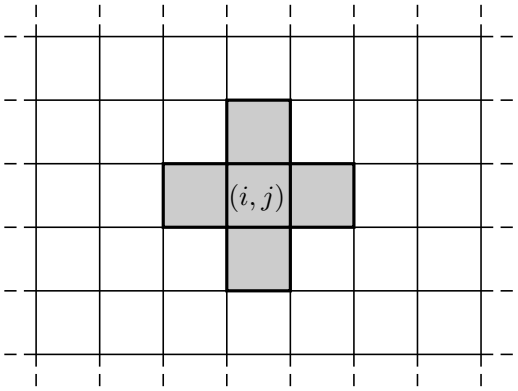


Figure 2.4: The von Neumann neighborhood of a cell (i, j) in a two-dimensional cellular space.

The domain $\mathcal{D}_2$ of this neighborhood is given by

$$\mathcal{D}_2 = \left\{ \begin{array}{c} x \\ y\ z\ u \\ v \end{array} \mid x, y, z, u, v \in Q \right\}.$$

The von Neumann neighborhood is one of the two most used two-dimensional neighborhoods. The second one is the *Moore neighborhood* presented in Figure 2.5 and used, for example, in the *Game of life*.

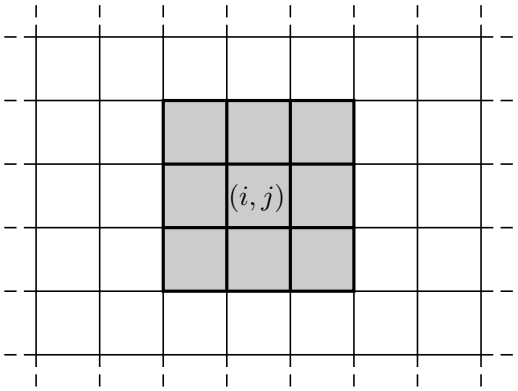


Figure 2.5: The Moore neighborhood of a cell (i, j) in a two-dimensional cellular space.

For brevity, we will denote the domain of the neighborhood for any dimension by $\mathcal{D}$, having in mind that for a two-dimensional CA we consider only the von Neumann neighborhood.

2.5 Rules

The state of every cell is updated at every discrete time step. We will denote the t-th time step of the CA evolution by t, thus $t \in \{0, 1, 2, \ldots\}$. The new state of a cell depends only on the states of the cells in its neighborhood at the previous time step. This dependency is encoded in a *local rule* $\mathbb{f} : \mathcal{D} \to Q$.

If the local rule $\mathbb{f}$ is given, we can define the *global rule* $\mathbb{F}$ acting on configurations as the result of the synchronous use of the local rule for all the cells. For the sake of clarity, for the one-dimensional CA and for $\boldsymbol{x} \in Q^n$, the global rule $\mathbb{F}$ is given by:

$$\mathbb{F}(\boldsymbol{x}) = \mathbb{F}(x_0, \ldots, x_{n-1}) = (x'_0, \ldots, x'_{n-1}),$$

where $x'_i = \mathbb{f}(x_{i-1}, x_i, x_{i+1})$.

For two-dimensional CAs, if $\boldsymbol{x}$ belongs to Q^{nm}, then $\mathbb{F}(\boldsymbol{x})$ also belongs to Q^{nm} and

$$\mathbb{F}(\boldsymbol{x})_{i,j} = \mathbb{f}\begin{pmatrix} & x_{i,j-1} & \\ x_{i-1,j} & x_{i,j} & x_{i+1,j} \\ & x_{i,j+1} & \end{pmatrix} .$$

Furthermore, for $t > 0$, the configuration $\mathbb{F}^t(\boldsymbol{x}) = \mathbb{F}(\mathbb{F}^{t-1}(\boldsymbol{x}))$ corresponds to the t-th time step of the CA evolution. Moreover, for a one-dimensional CA, we will write x_i^t to denote the value of cell i at the t-th time step in $\mathbb{F}^t(\boldsymbol{x})$. For two-dimensional CAs, we write $x_{i,j}^t$ for the value of cell (i, j).

2.6 LUT representation

If $Q = \{0, 1\}$, then a CA can be defined through its local rule $\mathbb{f}$ on $\{0, 1\}^{\mathcal{R}}$ by listing the $2^{\mathcal{R}}$ binary tuples $N_0, N_1, \ldots, N_s$, where $s = 2^{\mathcal{R}} - 1$, and mapping each tuple N_i to some $l_i \in \{0, 1\}$. The tuples $N_0, N_1, \ldots, N_s$ are arranged in such a way that N_i is the binary representation of the integer i, while $N_i(j)$ is the $(\mathcal{R} - j)$-th digit, from the left to the right of N_i, *i.e.*, we treat N_i as a vector of the form

$$N_i = (N_i(\mathcal{R} - 1), N_i(\mathcal{R} - 2), \ldots, N_i(1), N_i(0)) \in \{0, 1\}^{\mathcal{R}} , \tag{2.3}$$

such that the following holds:

$$i = N_i(\mathcal{R} - 1)2^{\mathcal{R}-1} + N_i(\mathcal{R} - 2)2^{\mathcal{R}-2} + N_i(1)2^1 + N_i(0)2^0 .$$

The sequence $(l_0, l_1, \ldots, l_s)$ will be referred to as the *lookup table* (*LUT*) of the local rule $\mathbb{f}$. For the one-dimensional cellular space, as there are only eight different neighborhood configurations, namely

$$\begin{aligned} N_0 &= (0, 0, 0) , \\ N_1 &= (0, 0, 1) , \\ N_2 &= (0, 1, 0) , \\ N_3 &= (0, 1, 1) , \\ N_4 &= (1, 0, 0) , \\ N_5 &= (1, 0, 1) , \\ N_6 &= (1, 1, 0) , \\ N_7 &= (1, 1, 1) , \end{aligned}$$

a local rule $\mathbb{f}$ can be defined by fixing the values $l_i = \mathbb{f}(N_i)$, for $i = 0, 1, \ldots, 7$. The final form of a LUT table for this one-dimensional cellular automata is presented in Table 2.1.

Table 2.1: General form of the LUT representation of the local rule with $Q = \{0, 1\}$, unit neighborhood radius ($r = 1$) and one-dimensional cellular space.

N_0	N_1	N_2	N_3	N_4	N_5	N_6	N_7
$(0, 0, 0)$	$(0, 0, 1)$	$(0, 1, 0)$	$(0, 1, 1)$	$(1, 0, 0)$	$(1, 0, 1)$	$(1, 1, 0)$	$(1, 1, 1)$
l_0	l_1	l_2	l_3	l_4	l_5	l_6	l_7

For two-dimensional binary cellular automata with the von Neumann

neighborhood, there are 32 different neighborhood configurations, where

$$N_i = \begin{matrix} & i_4 & \\ i_3 & i_2 & i_1 \\ & i_0 & \end{matrix}$$

and $i_4 i_3 i_2 i_1 i_0$ is the binary representation of the number i. In this case the sequence

$$(l_0, l_1, \ldots, l_{31}) \in \{0,1\}^{32}$$

will be referred to as the lookup table (LUT) of the two-dimensional local rule $\mathbb{f}$ (see Table 2.2).

Table 2.2: General form of the LUT of the local rule of a two-dimensional CA with the von Neumann neighborhood and $Q = \{0,1\}$.

0 0 0 0 0	0 0 0 0 1	0 0 0 1 0	0 0 0 1 1	0 0 1 0 0	0 0 1 0 1	0 0 1 1 0	0 0 1 1 1	0 1 0 0 0	0 1 0 0 1	0 1 0 1 0	0 1 0 1 1	0 1 1 0 0	0 1 1 0 1	0 1 1 1 0	0 1 1 1 1
l_0	l_1	l_2	l_3	l_4	l_5	l_6	l_7	l_8	l_9	l_{10}	l_{11}	l_{12}	l_{13}	l_{14}	l_{15}
1 0 0 0 0	1 0 0 0 1	1 0 0 1 0	1 0 0 1 1	1 0 1 0 0	1 0 1 0 1	1 0 1 1 0	1 0 1 1 1	1 1 0 0 0	1 1 0 0 1	1 1 0 1 0	1 1 0 1 1	1 1 1 0 0	1 1 1 0 1	1 1 1 1 0	1 1 1 1 1
l_{16}	l_{17}	l_{18}	l_{19}	l_{20}	l_{21}	l_{22}	l_{23}	l_{24}	l_{25}	l_{26}	l_{27}	l_{28}	l_{29}	l_{30}	l_{31}

The above representation reduces the risk of improper implementation in most programming languages, where arrays are indexed from zero as the neighborhood index N_i also points directly to the i-th index of the LUT array.

We will also recall two simple, but important definitions.

Definition 2.6.1 (Legal rule) *Let $\mathbb{f}$ be the local rule of a binary CA with radius $\mathcal{R}$ defined by the LUT $(l_0, \ldots, l_s)$, where $s = 2^{\mathcal{R}} - 1$. We say that the rule is legal if the LUT has a fixed entry $l_0 = 0$.*

In other words, a CA is called "legal" if it cannot produce anything from zeros, *i.e.*, its local rule $\mathbb{f}$ satisfies $l_0 = \mathbb{f}(0,0,0) = 0$ in the one-dimensional case and $l_0 = \mathbb{f}\begin{pmatrix} & 0 & \\ 0 & 0 & 0 \\ & 0 & \end{pmatrix} = 0$ in the two-dimensional case.

Definition 2.6.2 (Double-legal rule) *Let $\mathbb{f}$ be the local rule of a binary CA with radius $\mathcal{R}$ defined by the LUT $(l_0, \ldots, l_s)$, where $s = 2^{\mathcal{R}} - 1$. We say that the rule is double-legal if the LUT has fixed entries $l_0 = 0$ and $l_s = 1$.*

A double-legal rule is also known as a "*double-quiescent*" rule [45]. Both terms "legal" and "double-legal" are also used for other state sets Q, where $0 \in Q$ is required in both cases, and additionally $1 \in Q$ for "double-legal".

2.7 Rule symmetries

Although for a given Q and C there are many possible local rules, some of them are trivially equivalent to each other up to two kinds of simple

transformations. The first ones are based on neighborhood symmetries, while the second ones take into account the conjugation of states.

In the one-dimensional case, there are only three such transformations:

- inversion of the neighborhood – obtained by the exchange of the left and right directions,
- conjugation of the states – for example in the case of binary CAs, obtained by the exchange of states 0 and 1,
- composition of both above transformations.

Formally, let $\mathbb{f}$ be the local rule of a one-dimensional CA with radius one and let $Q = Q_B$ or $Q = Q_{\mathbb{R}}$. The inversion and the conjugation of $\mathbb{f}$ are defined as follows:

$$\mathbb{f}^I(x, y, z) = \mathbb{f}(z, y, x)\,, \tag{2.4}$$

$$\mathbb{f}^C(x, y, z) = 1 - \mathbb{f}(1 - x, 1 - y, 1 - z)\,, \tag{2.5}$$

for $x, y, z \in Q$. Note that both of these transformations are involutions, *i.e.*, $(\mathbb{f}^I)^I = \mathbb{f}$ and $(\mathbb{f}^C)^C = \mathbb{f}$. Moreover, I commutes with C, *i.e.*, for any local rule $\mathbb{f}$ it holds that

$$\mathbb{f}^{IC} := (\mathbb{f}^I)^C = (\mathbb{f}^C)^I =: \mathbb{f}^{CI}\,.$$

An example of the results of the transformations I, C and IC is presented in Figure 2.6.

Now we describe the rule symmetries in the case of two dimensions. We only focus on the von Neumann neighborhood and $Q = Q_B$ or $Q = Q_{\mathbb{R}}$. We can rely on the eight symmetries of the von Neumann neighborhood (along a vertical, horizontal or diagonal axis and rotation by 0, 90, 180 or 270 degrees) and possibly the conjugation operator C (exchanging states 0 and 1):

$$\mathbb{f}^C(\boldsymbol{z}) = \mathbb{f}^C\begin{pmatrix} & x & \\ y & z & u \\ & v & \end{pmatrix} = 1 - \mathbb{f}\begin{pmatrix} & 1-x & \\ 1-y & 1-z & 1-u \\ & 1-v & \end{pmatrix} = 1 - \mathbb{f}(\mathbf{1} - \boldsymbol{z})\,. \tag{2.6}$$

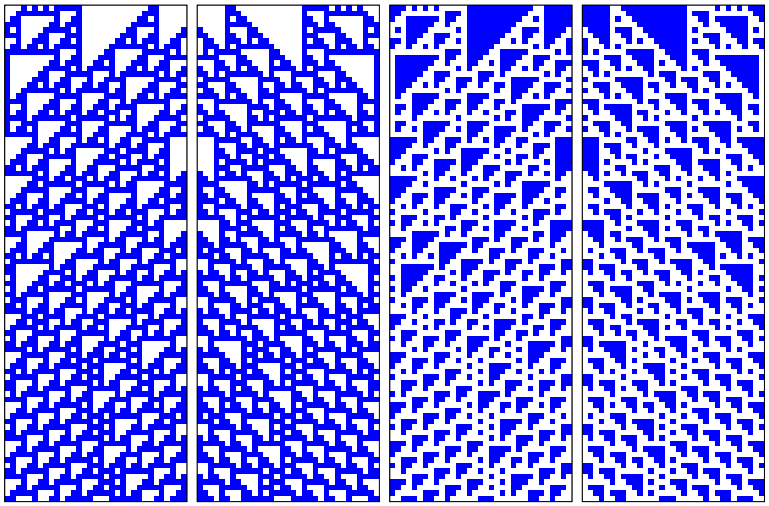
Thus, we have as many as 16 transformations that form a group, which is homeomorphic with the direct product $\mathcal{T} = D_4 \times \{\mathrm{Id}, C\}$, where D_4 is the group of symmetries of the square and $\{\mathrm{Id}, C\}$ is a cyclic group of order two. The group $\mathcal{T}$ can be generated by three transformations:

- I: axial flip along the vertical bisector of the von Neumann neighborhood,
- R: clockwise rotation by 90 degrees of the von Neumann neighborhood,
- C: conjugation operator.

The transformations I and C are involutions, while R is of order four. Moreover, C commutes with both I and R. In the case of one dimension, for any transformation $T \in \mathcal{T}$, the behaviors of CAs with the local rules $\mathbb{f}$ and $\mathbb{f}^T$ are strongly related.

2.8 Elementary Cellular Automata

The simplest one-dimensional CAs with a binary state set Q_B and neighborhood with radius $r = 1$ are called Elementary CAs (*ECAs*) and were extensively studied by Stephen Wolfram [12]. ECA rules can be indexed using a decimal representation of a binary string created from the LUT, namely $l_7 l_6 l_5 l_4 l_3 l_2 l_1 l_0$. This results in only 256 ECA rules from ECA0 to ECA255. Note that the binary representation is the reverse order of the LUT. Moreover, taking into account the rule symmetries given in Eqs. (2.4) and (2.5), we have only 88 unique ECA rules. An example is rule ECA110, which has three equivalent rules ECA124 ($\mathbb{f}^I$), ECA137 ($\mathbb{f}^C$) and ECA193 ($\mathbb{f}^{IC}$). These rules can be visualized in a space-time diagram (Figure 2.6) where time flows downwards. Each t-th line in the



(a) $\mathbb{f}$ (ECA110) **(b)** $\mathbb{f}^I$ (ECA124) **(c)** $\mathbb{f}^C$ (ECA137) **(d)** $\mathbb{f}^{IC}$ (ECA193)

Figure 2.6: Space-time diagrams of rule $\mathbb{f}$:ECA110, starting from configuration $\boldsymbol{x} = 0^3(10)^3 1^5 0^{12} 1^2 0^5$, and its equivalent rules $\mathbb{f}^I$, $\mathbb{f}^C$ and $\mathbb{f}^{IC}$ starting on a mirrored, negated and mirrored-negated configuration $\boldsymbol{x}$, respectively.

diagram represents the result of the t-th time step of the evolution of the configuration $\boldsymbol{x}$, *i.e.*, line 0 on top represents the configuration $\boldsymbol{x}$, and t-th line represents $\mathbb{F}^t(\boldsymbol{x})$.

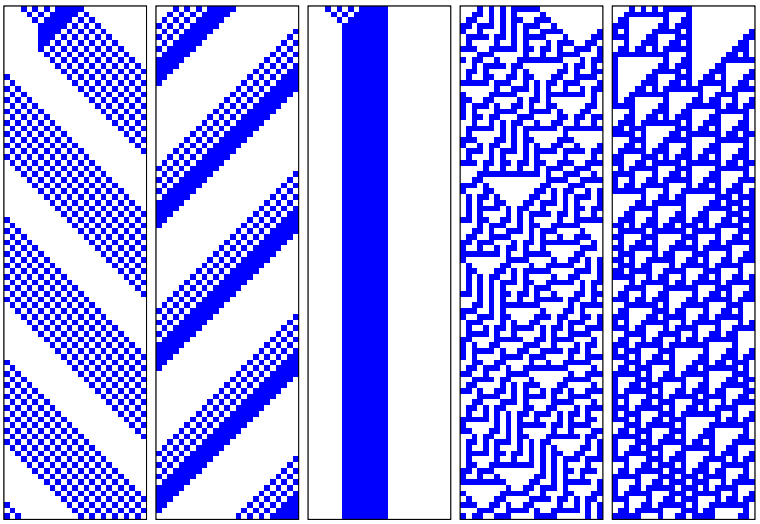
Wolfram, based on an experimental study, provided a general classification of ECAs. This classification is based on observations of the space-time diagram, the behavior of the states, and is numbered in order of growing complexity. We will describe a few ECAs from each class presented in Table 2.3; some ECAs will also be mentioned in our further considerations.

ECA	Class	Name	000	001	010	011	100	101	110	111
			l_0	l_1	l_2	l_3	l_4	l_5	l_6	l_7
0	I	-	0	0	0	0	0	0	0	0
30	III	-	0	1	1	1	1	0	0	0
51	II	negation	1	1	0	0	1	1	0	0
110	IV	-	0	1	1	1	0	1	1	0
170	II	shift-left	0	1	0	1	0	1	0	1
184	II	traffic-right	0	0	1	0	1	1	0	1
204	II	identity	0	0	1	1	0	0	1	1
226	II	traffic-left	0	1	0	0	0	1	1	1
232	II	majority	0	0	0	1	0	1	1	1
240	II	shift-right	0	0	0	0	1	1	1	1
255	I	-	1	1	1	1	1	1	1	1

Table 2.3: Selected ECA rules with their LUT representation and Wolfram class.

Rules from *Class I* have a very simple and uninteresting behavior. All initial configurations evolve to the same homogeneous state, *i.e.*, to a configuration with the same value in each cell. In this class, there are only eight unique ECAs and ECA0 is an example rule from this class, where each initial configuration evolves to 0. The equivalent rule is ECA255, where each configuration evolves to 255.

In *Class II*, initial configurations evolve to periodic structures that persist or repeat every few iterations. In this class, there are 65 unique rules. The *identity rule*, ECA204, is an example rule that preserves the states of the initial configuration, *i.e.*, every iteration is unchanged. Rule ECA51, called the *negation* rule, is another example of this class. This rule negates states to the opposite state, and as a result, repeats every second iteration. Another example is the *shift-left* rule ECA170. This rule shifts the cell values to the left and can be seen in the space-time diagram in Figure 2.7b. The equivalent rule is ECA240, the *shift-right*, which does the same in the other direction. Such rules, which only move the cell value to another cell, will be called *shift rules*. Another example is rule ECA184 that can be used as a simple model for the traffic flow, and for this reason the name *traffic-rule* is used to identify the rule. In some situations, like with shift rules, we will require the specific behavior of the rule and use the name *traffic-right* for rule ECA184, as the flow goes from left to right, and for its equivalent ECA226, *traffic-left*. The space-time diagram of the *traffic-right* rule can be seen in Figure 2.7a.



(a) ECA184 **(b)** ECA170 **(c)** ECA232 **(d)** ECA30 **(e)** ECA110

Figure 2.7: Space-time diagrams of selected ECAs for the same initial configuration $\boldsymbol{x} = (0)^3(10)^3(1)^5(0)^{11}$.

Class III has 11 unique rules and describes rules with more complicated behavior and aperiodic patterns. An example of a rule from this class is rule ECA30. This rule was already mentioned in Section 1.3 as a pseudo-random number generator. The space-time diagram of this rule is presented in Figure 2.7d.

The last *Class IV* generates complex patterns and has only four unique rules, but on the other hand, with a very interesting behavior. An example is ECA110, with the space-time diagram in Figure 2.7e. This rule is the only one known (excluding its inversion ECA124 and conjugation ECA127) to be Turing complete with a proof given by Cook [11].

2.9 Continuous Cellular Automata

Continuous cellular automata (*CCA*s) can be seen as a generalization of cellular automata (CAs), in which we keep the space and time discrete, but we accept that cell states can be taken from some infinite set. In general, the state set of a CCA can be any subset of $\mathbb{R}$, but in this work, we assume that the CCA states belong to the unit interval $[0, 1]$.

A well-known generalization of binary CAs to CCAs with $[0, 1]$ as the state set is obtained by "*fuzzification*" in [46] and was further studied, for example, in [47, 48]. This process allows to construct some particular CCA (a so-called *Fuzzy CA*) for every binary CA through the extension of the domain of the local rule. The relationship between Fuzzy CAs and ECAs can be found in [49].

The diffusion rule is a good example of a CCA rule. We define the local rule $\mathbb{f}_d$ for a one-dimensional CCA:

$$\mathbb{f}_d(x, y, z) = \frac{1}{3}(x + y + z) \,. \tag{2.7}$$

For two-dimensional CCAs with the von Neumann neighborhood, the rule $\mathbb{f}_d$ is defined analogously

$$\mathbb{f}_d\left(\begin{smallmatrix} & x & \\ y & z & u \\ & v & \end{smallmatrix}\right) = \frac{1}{5}(x + y + z + u + v) \,. \tag{2.8}$$

This rule states that the new state of any cell is given by the average of its own state and those of its neighboring cells. A space-time diagram of the one-dimensional diffusion rule is presented in Figure 2.8.

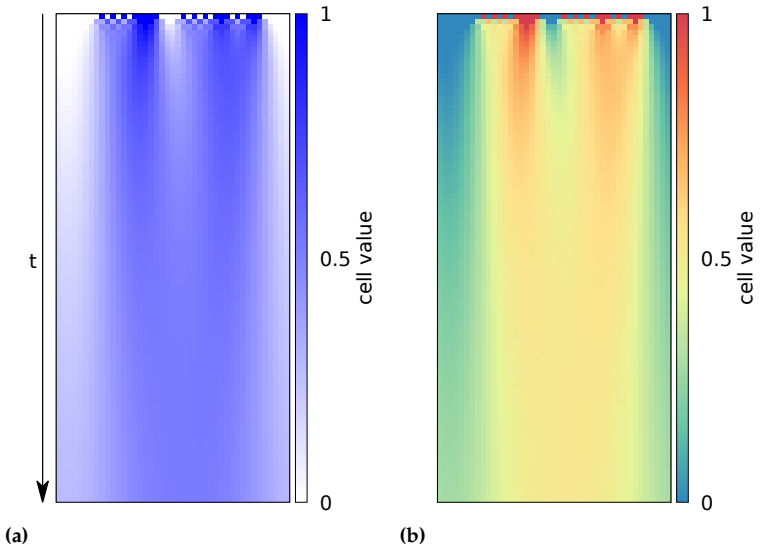
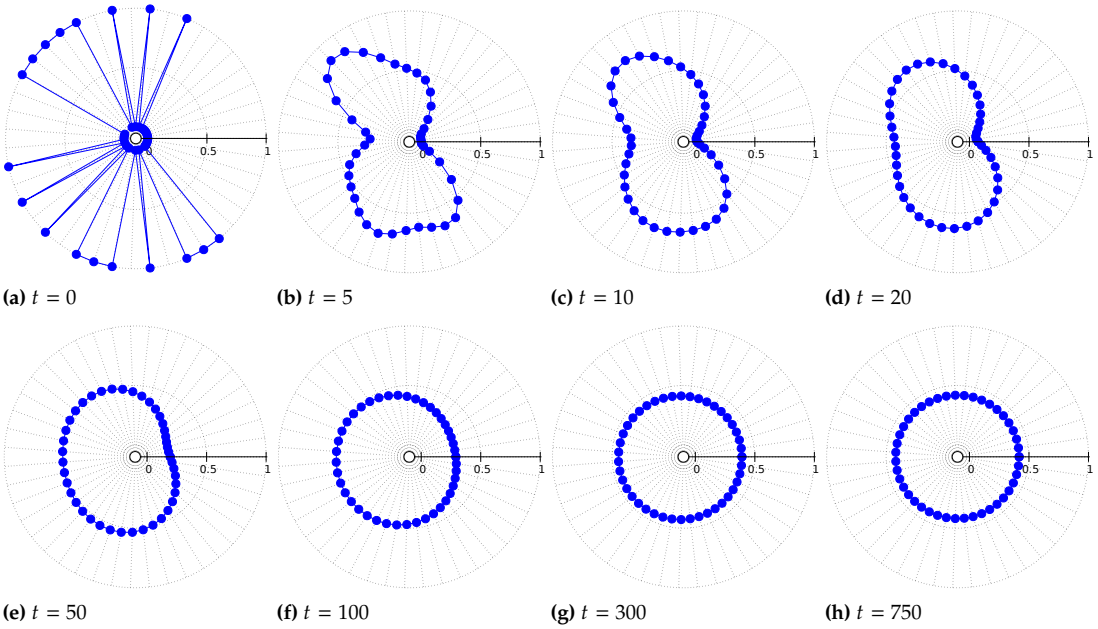


Figure 2.8: Space-time diagram of the one-dimensional diffusion CCA rule $\mathbb{f}_d$ for a configuration x and different cell value palette representations: (a) blue-white gradient and (b) multicolor.

For binary CAs (ECAs in particular) the binary palette is sufficient to visualize the observations in space-time diagrams. For CCAs, the most commonly used method is the extension of the palette where each cell value is represented by its own color or gradient. In Figure 2.8a a blue-white gradient is used to visualize the cell values from 1 to 0 and in Figure 2.8b a multicolor palette is used, where each cell value has its own color. For CCAs it might be difficult to observe dynamics from the

diagram for a specific iteration, not only because there is a limitation of the finite color space to maximum 2^{32} colors, but also by the limitation of the human eye.

To visualize the dynamics of a CCA, we will use a slight modification of the radial view of CCAs – a method proposed in [50]. For a one-



(a) $t = 0$ **(b)** $t = 5$ **(c)** $t = 10$ **(d)** $t = 20$

(e) $t = 50$ **(f)** $t = 100$ **(g)** $t = 300$ **(h)** $t = 750$

Figure 2.9: A radial view representation of a configuration $\boldsymbol{x}$ in different iterations t of the one-dimensional diffusion CCA rule $\mathbb{f}_d$.

dimensional CCA, if the number of cells N is given, we consider a circle with radius 1.1 and center C divided in N equal sectors by points A_0, $A_1, \ldots, A_{N-1}$. The value x_n of the n-th cell is represented by a dot lying on the line segment CA_n at a distance $0.1 + x_n$ from C. For the sake of convenience, we connect subsequent dots using a polyline. Examples of the radial representation of a configuration from the space-time diagram in Figure 2.8 for selected iterations are presented in Figure 2.9.

Note that in the literature, CCAs are often referred to as Coupled Map Lattices (CMLs) [51]. Yet, in most studies on CMLs, only specific local rules are considered, typically taking a linear combination of the inputs individually transformed by means of a given (non-)linear function. The local rules considered later in this work are not of this type.

2.10 Conclusions

In this chapter, we provided formal definitions for one- and two- dimensional CAs. Due to their simplicity, CAs can be easily implemented in any modern programming language without any sophisticated mathematical libraries. In some cases, like for the initial configurations or rule symmetries, the amount of necessary computations can be reduced. In the next chapter, we will extend some properties of CCAs.

Affine Continuous Cellular Automata

3

In this chapter, we give the formal definitions of Affine Continuous Cellular Automata (*ACCA*s), an extension of CCAs.

3.1 Preliminaries 19
3.2 One-dimensional ACCAs 20
3.3 Outer-totalistic one-dimensional ACCAs . . . 20
3.4 Density-conserving one-dimensional ACCAs 22
3.5 Two-dimensional ACCAs 26
3.6 Density-conserving two-dimensional ACCAs 26
3.6.1 Preliminaries 26
3.6.2 Structure of the space of density-conserving two-dimensional ACCAs 28
3.6.3 Symmetries in the space of density-conserving two-dimensional ACCAs 29
3.6.4 Density-conserving two-dimensional ACCAs with independent y and v . . . 30
3.7 Rotation-symmetric two-dimensional ACCAs 30
3.8 Conclusions 32

3.1 Preliminaries

The term "*Affine*" in the name of these CCAs means that the local rule $\mathbb{f}$ of such CA is affine in each variable.

Recall that a function $f(x_1, \ldots, x_k)$ is affine in the variable x_i when, for any fixed $x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_k$, the function $f_i\colon [0,1] \to [0,1]$ given by $f_i(x_i) = f(x_1, \ldots, x_k)$ is affine, *i.e.*, $f_i(x_i) = ax_i + b$ for some $a, b \in \mathbb{R}$. This kind of CCAs was considered earlier as a product of "*fuzzification*" of ECAs [48, 49, 52]. Given the fact that a function that is affine in some variable, must be monotone in it, we get the following obvious but important fact [49] presented in Theorem 3.1.1. This theorem indicates that both the minimal and maximal value of the local rule of an ACCA belong to the set of LUT values.

Theorem 3.1.1 *Let $0 \le m \le M \le 1$. If a function $f\colon [0,1]^k \to [0,1]$ is affine in each variable, then, for any $\boldsymbol{x} \in [m,M]^k$, it holds that:*

$$\min\left\{f(\boldsymbol{y}) \mid \boldsymbol{y} \in \{m,M\}^k\right\} \le f(\boldsymbol{x}) \le \max\left\{f(\boldsymbol{y}) \mid \boldsymbol{y} \in \{m,M\}^k\right\} . \tag{3.1}$$

Below, we present the proof of Theorem 3.1.1 in the case $k = 3$ but the idea of the proof for an arbitrary k is the same.

Proof. Let us denote $L = \min\left\{f(y) \mid y \in \{m,M\}^3\right\}$. First, as f is affine in the first variable, then for any fixed $y, z \in [0,1]$ the function $f_{y,z} : [0,1] \to [0,1]$ defined by $f_{y,z}(\boldsymbol{x}) = f(x,y,z)$ is monotone, thus its value at any point of the considered interval $[m,M]$ lies between the values at the endpoints of this interval. So it holds that for any $x \in [m,M]$:

$$\begin{aligned} f(x,m,m) &\ge \min\left(f(m,m,m), f(M,m,m)\right) \ge L , \\ f(x,m,M) &\ge \min\left(f(m,m,M), f(M,m,M)\right) \ge L , \\ f(x,M,m) &\ge \min\left(f(m,M,m), f(M,M,m)\right) \ge L , \\ f(x,M,M) &\ge \min\left(f(m,M,M), f(M,M,M)\right) \ge L . \end{aligned} \tag{3.2}$$

Secondly, as f is affine in the second variable, for any $x, y \in [m,M]$ according to Eq. (3.2), it holds that

$$\begin{aligned} f(x,y,m) &\ge \min\left(f(x,m,m), f(x,M,m)\right) \ge L , \\ f(x,y,M) &\ge \min\left(f(x,m,M), f(x,M,M)\right) \ge L . \end{aligned} \tag{3.3}$$

Finally, we use the fact that f is affine in the third variable, thus for any $x, y, z \in [m, M]$, we have from Eq. (3.3)

$$f(x, y, z) \geq \min\left(f(x, y, m), f(x, y, M)\right) \geq L \ .$$

This finishes the proof of the left inequality in Eq. (3.1), *i.e.*, the lower bound. The proof of the upper bound is analogous. □

3.2 One-dimensional ACCAs

To define the local rule $\mathbb{f}$ of a one-dimensional ACCA with radius r, we start by defining the values of $\mathbb{f}$ on $\{0,1\}^{\mathcal{R}}$, where $\mathcal{R} = 2r + 1$, *i.e.*, we fix the values $l_i \in [0, 1]$. Because we want the function $\mathbb{f}$ to be affine in each variable x_i, there is a unique extension of $\mathbb{f}$ from $\{0,1\}^{\mathcal{R}}$ to the entire $[0,1]^{\mathcal{R}}$, which can be expressed as a polynomial:

$$\mathbb{f}(x_0, \ldots, x_{\mathcal{R}-1}) = \sum_{i=0}^{s} l_i \left(\prod_{k=0}^{\mathcal{R}-1} h(x_k, N_i(\mathcal{R} - 1 - k)) \right) , \qquad (3.4)$$

where $h(x_k, 1) = x_k$, $h(x_k, 0) = 1 - x_k$ and $N_i(j)$ is defined by Eq. (2.3).

Example 3.2.1 If $\mathcal{R} = 3$ (*i.e.*, $r = 1$), then the cardinality of $\{0,1\}^{\mathcal{R}}$ is eight, so the corresponding LUT can be presented in tabular form (Table 2.1). Besides, evaluating Eq. (3.4) gives:

$$\begin{aligned}\mathbb{f}(x, y, z) =& l_0(1-x)(1-y)(1-z) + l_1(1-x)(1-y)z \\ &+ l_2(1-x)y(1-z) + l_3(1-x)yz + l_4x(1-y)(1-z) \\ &+ l_5x(1-y)z + l_6xy(1-z) + l_7xyz \ ,\end{aligned}$$

or, in its canonical form:

$$\begin{aligned}\mathbb{f}(x, y, z) =& (l_7 - l_6 - l_5 - l_3 + l_4 + l_2 + l_1 - l_0)xyz \\ &+ (l_6 - l_4 - l_2 + l_0)xy + (l_5 - l_4 - l_1 + l_0)xz \\ &+ (l_3 - l_2 - l_1 + l_0)yz + (l_4 - l_0)x \\ &+ (l_2 - l_0)y + (l_1 - l_0)z + l_0 \ .\end{aligned}$$

In particular, for the LUT $\left(0, \frac{1}{3}, \frac{1}{3}, \frac{2}{3}, \frac{1}{3}, \frac{2}{3}, \frac{2}{3}, 1\right)$, we get the diffusion rule $\mathbb{f}_d$ from Eq. (2.7).

3.3 Outer-totalistic one-dimensional ACCAs

Let us recall the definition of outer-totalistic CAs.

Definition 3.3.1 *Let $\mathbb{f}$ be the local rule of a CA. We say that a CA is outer-totalistic when the local rule $\mathbb{f}$ depends only on the state of the focal cell and the sum of the states of the adjacent cells.*

The next theorem gives a necessary and sufficient condition for one-dimensional ACCA to be outer-totalistic.

Theorem 3.3.1 *A one-dimensional ACCA with unit neighborhood radius is outer-totalistic if and only if the entries of the LUT of its local rule satisfy*

$$\begin{cases} l_0 = a \ , \\ l_1 = b \ , \\ l_2 = c \ , \\ l_3 = d \ , \\ l_4 = b \ , \\ l_5 = 2b - a \ , \\ l_6 = d \ , \\ l_7 = 2d - c \ , \end{cases} \tag{3.5}$$

where $a, b, c, d \in [0,1]$, $\frac{1}{2}a \leq b \leq \frac{1}{2}a + \frac{1}{2}$ *and* $\frac{1}{2}c \leq d \leq \frac{1}{2}c + \frac{1}{2}$. *In other words, the local rule* $\mathbb{f}$ *of an outer-totalistic ACCA is given by the following formula*

$$\mathbb{f}(x,y,z) = (d-c-b+a)(x+z)y + (b-a)(x+z) + (c-a)y + a \ . \tag{3.6}$$

Proof. It is easy to see that if $\mathbb{f}$ is given by Eq. (3.6), then the corresponding ACCA is outer-totalistic. To prove that Eq. (3.5) is necessary, it suffices to note that the following equalities have to hold:

$$\begin{cases} \mathbb{f}(1,0,0) = \mathbb{f}\left(\frac{1}{2}, 0, \frac{1}{2}\right) = \mathbb{f}(0,0,1) \ , \\ \mathbb{f}(1,1,0) = \mathbb{f}\left(\frac{1}{2}, 1, \frac{1}{2}\right) = \mathbb{f}(0,1,1) \ , \end{cases}$$

so, from Eq. (3.4) we get

$$\begin{cases} l_4 = \frac{1}{4}(l_0 + l_1 + l_4 + l_5) = l_1 \ , \\ l_6 = \frac{1}{4}(l_2 + l_3 + l_6 + l_7) = l_3 \ , \end{cases}$$

which gives Eq. (3.5). Substituting Eq. (3.5) into Eq. (3.4), we get Eq. (3.6). □

Let us note that for Fuzzy CAs the necessary and sufficient condition for being outer-totalistic is exactly the same as for ACCAs in Theorem 3.3.1 with the additional assumption that $a, b, c, d \in \{0,1\}$. As a consequence, we obtain that among the 64 outer-totalistic ECAs, only four of them are still outer-totalistic when we extend the domain of their local rules from $\{0,1\}^3$ to $[0,1]^3$. These are: ECA0, ECA51, ECA204 and ECA255 (see Table 2.3). Thus, there are only four outer-totalistic Fuzzy CAs with the following local rules:

$$\begin{aligned} &\mathbb{f}_1(x,y,z) = 0 \ , \\ &\mathbb{f}_2(x,y,z) = 1 - y \ , \\ &\mathbb{f}_3(x,y,z) = y \ , \\ &\mathbb{f}_4(x,y,z) = 1 \ . \end{aligned}$$

All these rules give rise to a very simple dynamics.

According to Theorem 3.3.1, the set of outer-totalistic ACCAs with unit neighborhood radius can be parameterized by points of the four-dimensional polytope, which can be obtained as a Cartesian product $D \times D$, where the parallelogram D has vertices $(0,0)$, $\left(1,\frac{1}{2}\right)$, $(1,1)$ and $\left(0,\frac{1}{2}\right)$ (Figure 3.1).

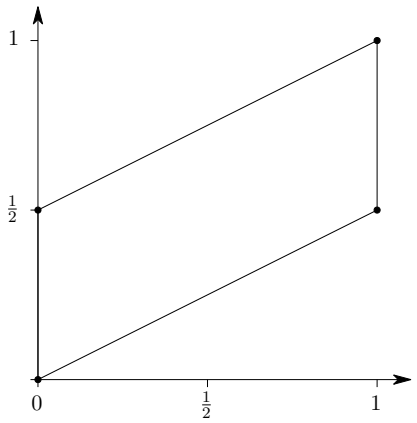


Figure 3.1: Parallelogram D that creates a four-dimensional polytope $D \times D$ of the set of outer-totalistic ACCAs.

It is easy to see that if some ACCA is outer-totalistic and its local rule $\mathbb{f}$ is parameterized by a point (a,b,c,d), then the conjugate CA, whose local rule $\mathbb{f}^C$ is given by Eq. (2.5), is also an outer-totalistic ACCA. Moreover, $\mathbb{f}^C$ is parameterized by the point (a',b',c',d'), where $a' = 1+c-2d$, $b' = 1-d$, $c' = 1+a-2b$ and $d' = 1-b$. Note that the behavior of $\mathbb{F}^C$ is strongly related to the behavior of $\mathbb{F}$. Indeed, if $\boldsymbol{x}$ is any configuration from $[0,1]^N$ and $\boldsymbol{y}$ is the configuration given by $y_i = 1 - x_i$, for $i \in \{0,1,\ldots,N-1\}$, then at each time step $t \geq 0$ it holds that $y_i^t = 1 - x_i^t$. For binary CAs the conjugation $\mathbb{F} \mapsto \mathbb{F}^C$ is often called the white-to-black transformation.

3.4 Density-conserving one-dimensional ACCAs

In this section, we define density-conserving one-dimensional ACCAs. For the sake of simplicity, we restrict to ACCAs with $r = 1$.

Definition 3.4.1 (Density-conserving CCA) *Let $\mathbb{F}$ be the global rule of a CCA. We say that this CCA is* density-conserving *if for any $\boldsymbol{x} \in [0,1]^*$, it holds that:*

$$\rho(\boldsymbol{x}) = \rho(\mathbb{F}(\boldsymbol{x})) \ .$$

In other words, a density-conserving ACCA conserves the sum of states.

Theorem 3.4.1 gives a full characterization of density-conserving one-dimensional ACCAs in the case of unit neighborhood radius.

Theorem 3.4.1 *Let $\mathbb{f}$ be the local rule of a one-dimensional ACCA with unit neighborhood radius. Then the ACCA conserves density if and only if for all $(x,y,z) \in \{0,1\}^3$ it holds that*

$$\mathbb{f}(x,y,z) = x + [\mathbb{f}(0,y,z) - \mathbb{f}(0,x,y))] + [\mathbb{f}(0,0,y) - \mathbb{f}(0,0,x)] \ . \quad (3.7)$$

Proof. The proof for ECAs can be found in [53]. Here, we extend it to continuous case. We start with the proof that the condition in Eq. (3.7) is necessary. First, let us consider a null configuration $(0,0,\ldots,0) \in \{0,1\}^\star$. Of course, the density of $(0,0,\ldots,0)$ equals 0. Furthermore, $\mathbb{F}(0,0,\ldots,0) = (c,c,\ldots,c)$, where $c = \mathbb{f}(0,0,0)$, so the density of $\mathbb{F}(0,0,\ldots,0)$ equals c. Thus, if the ACCA conserves density, then it holds that $c = 0$. Now, let $x,y,z \in \{0,1\}$ and let us consider the configurations:

$$\boldsymbol{x} = (0,0,\ldots,0,x,y,z,0,\ldots,0) \in \{0,1\}^N$$

and

$$\boldsymbol{y} = (0,0,\ldots,0,0,y,z,0,\ldots,0) \in \{0,1\}^N \ ,$$

where $N \geq 7$. Note that

$$\begin{array}{c} (0,\ldots,0,x,y,z,0,\ldots,0) \\ \Big\downarrow \mathbb{F} \\ (0,\ldots,\mathbb{f}(0,0,x),\mathbb{f}(0,x,y),\mathbb{f}(x,y,z),\mathbb{f}(y,z,0),\mathbb{f}(z,0,0),0,\ldots,0) \ , \end{array}$$

while

$$\begin{array}{c} (0,0,\ldots,0,0,y,z,0,\ldots,0) \\ \Big\downarrow \mathbb{F} \\ (0,0,\ldots,0,\mathbb{f}(0,0,y),\mathbb{f}(0,y,z),\mathbb{f}(y,z,0),\mathbb{f}(z,0,0),0,\ldots,0) \ . \end{array}$$

Using the fact that if the ACCA conserves density, then the sum of the states for $\boldsymbol{x}$ is the same as for $\mathbb{F}(\boldsymbol{x})$, and the sum of the states for $\boldsymbol{y}$ is the same as for $\mathbb{F}(\boldsymbol{y})$, we get two equations:

$$x + y + z = \mathbb{f}(0,0,x) + \mathbb{f}(0,x,y) + \mathbb{f}(x,y,z) + \mathbb{f}(y,z,0) + \mathbb{f}(z,0,0) \quad (3.8)$$

and

$$y + z = \mathbb{f}(0,0,y) + \mathbb{f}(0,y,z) + \mathbb{f}(y,z,0) + \mathbb{f}(z,0,0) \ . \quad (3.9)$$

Subtracting Eq. (3.9) from Eq. (3.8) we get

$$x = \mathbb{f}(0,0,x) - \mathbb{f}(0,0,y) + \mathbb{f}(0,x,y) - \mathbb{f}(0,y,z) + \mathbb{f}(x,y,z) \ ,$$

which gives Eq. (3.7). To prove that the condition in Eq. (3.7) is sufficient, let us suppose that $\boldsymbol{x} = (x_0, x_1, \ldots, x_{N-1}) \in \{0,1\}^N$. Then

$$\begin{array}{c} (x_0, x_1 \ , \ldots , \ x_{N-1}) \\ \Big\downarrow \mathbb{F} \\ (\mathbb{f}(x_{N-1},x_0,x_1), \mathbb{f}(x_0,x_1,x_2) \ , \ldots , \ \mathbb{f}(x_{N-2},x_{N-1},x_0)) \ . \end{array}$$

According to Eq. (3.7)

$$\begin{aligned} \mathbb{f}(x_{N-1},x_0,x_1) + \mathbb{f}(x_0,x_1,x_2) + \ldots + \mathbb{f}(x_{N-2},x_{N-1},x_0) \\ = x_0 + x_1 + \ldots + x_{N-1} \ , \end{aligned} \quad (3.10)$$

since when summing Eq. (3.7) on the entire configuration, all the right-hand side terms except the first cancel. Finally, let us recall that according to [48], if Eq. (3.10) holds for any $\boldsymbol{x} \in \{0,1\}^N$, then it holds also for any $\boldsymbol{x} \in [0,1]^N$. □

Corollary 3.4.2 *Let $\mathbb{f}$ be the local rule of an ACCA with unit neighborhood radius and $l_0, \dots, l_7 \in [0,1]$ be the entries in its LUT. Then the ACCA conserves density if and only if:*

$$\begin{cases} l_0 = 0 \,, \\ l_1 + l_2 + l_4 = 1 \,, \\ l_2 + l_5 = 1 \,, \\ l_3 + l_5 + l_6 = 2 \,, \\ l_7 = 1 \,. \end{cases} \tag{3.11}$$

Proof. According to the above theorem, the density conservation of the ACCA is equivalent with the following system of equations:

$$\begin{cases} \mathbb{f}(0,0,0) = 0 \,, \\ \mathbb{f}(0,0,1) = \mathbb{f}(0,0,1) \,, \\ \mathbb{f}(0,1,0) = \mathbb{f}(0,1,0) \,, \\ \mathbb{f}(0,1,1) = \mathbb{f}(0,1,1) \,, \\ \mathbb{f}(1,0,0) = 1 - \mathbb{f}(0,1,0) - \mathbb{f}(0,0,1) \,, \\ \mathbb{f}(1,0,1) = 1 - \mathbb{f}(0,1,0) \,, \\ \mathbb{f}(1,1,0) = 1 + \mathbb{f}(0,1,0) - \mathbb{f}(0,1,1) \,, \\ \mathbb{f}(1,1,1) = 1 \,. \end{cases}$$

This means in terms of $l_0, l_1, \dots, l_7$

$$\begin{cases} l_0 = 0 \,, \\ l_4 = 1 - l_2 - l_1 \,, \\ l_5 = 1 - l_2 \,, \\ l_6 = 1 + l_2 - l_3 \,, \\ l_7 = 1 \,. \end{cases} \tag{3.12}$$

Thus substituting in the fourth line $l_2 = 1 - l_5$, we get the claim. □

A similar result is also reported for Stochastic CAs [54], and interestingly, the same system of equations is obtained there.

As a consequence of Theorem 3.4.1, the local rule $\mathbb{f}$ of a density-conserving ACCA with $r = 1$ can be written as (cfr. Example 3.2.1):

$$\mathbb{f}(x,y,z) = (l_6 + l_1 - 1)(x - z)y + l_4 x + l_2 y + l_1 z \,. \tag{3.13}$$

The solutions of System (3.12) are of the following form:

$$(l_0, l_1, l_2, l_3, l_4, l_5, l_6, l_7) = (0, a, c, 1-b, 1-a-c, 1-c, b+c, 1) \,, \tag{3.14}$$

where $a, b, c \in [0,1]$ and $a, b \leq 1 - c$. The solution set thus depends on three parameters and can be represented graphically in the Cartesian coordinate system as a pyramid P (Figure 3.2). In particular, the local rule $\mathbb{f}_d$ (Eq. (2.7)) is represented in the pyramid P as the point

$$(a,b,c) = \left(\frac{1}{3}, \frac{1}{3}, \frac{1}{3}\right).$$

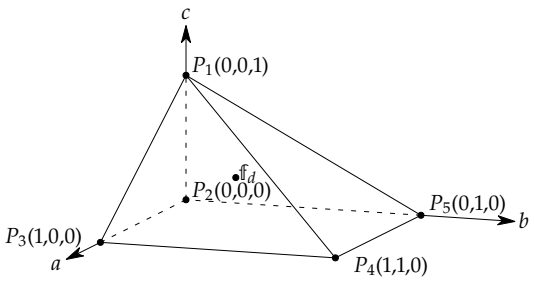


Figure 3.2: The pyramid P: parameterization of the set of one-dimensional, density-conserving ACCAs with unit neighborhood radius.

Taking into account the symmetry of ACCAs, it is easy to see that if $\mathbb{f}$ defines an ACCA that conserves density, also $\mathbb{f}^I, \mathbb{f}^C$ and $\mathbb{f}^{IC}$ define density-conserving ACCAs, and the behavior of these ACCAs is strongly related. Let $\pi_{\mathbb{f}}$ denote the point in the pyramid P shown in Figure 3.2,

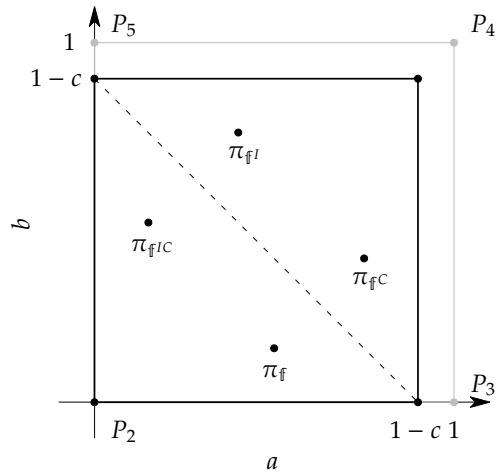


Figure 3.3: The location of the points $\pi_{\mathbb{f}}, \pi_{\mathbb{f}^I}, \pi_{\mathbb{f}^C}$ and $\pi_{\mathbb{f}^{IC}}$ corresponding to the local rules $\mathbb{f}, \mathbb{f}^I, \mathbb{f}^C$ and $\mathbb{f}^{IC}$.

and representing the local rule $\mathbb{f}$. Then the local rules $\mathbb{f}^I, \mathbb{f}^C$ and $\mathbb{f}^{IC}$ are represented by the points $\pi_{\mathbb{f}^I}$, $\pi_{\mathbb{f}^C}$ and $\pi_{\mathbb{f}^{IC}}$ lying in the same square cross-section obtained by cutting P with a plane parallel to its base. Moreover, $\pi_{\mathbb{f}^I}$ is symmetric to $\pi_{\mathbb{f}}$ w.r.t. the center of the square, while $\pi_{\mathbb{f}^C}$ is symmetric to $\pi_{\mathbb{f}}$ w.r.t. the main diagonal (Figure 3.3). The apex $P_1(0,0,1)$ of the pyramid corresponds to ECA204, while the vertices $P_2(0,0,0), P_3(1,0,0), P_4(1,1,0), P_5(0,1,0)$ of its base correspond to the following ECAs: ECA184, ECA170, ECA226 and ECA240. Since the dynamics of the ACCAs defined by $\mathbb{f}, \mathbb{f}^I, \mathbb{f}^C$ and $\mathbb{f}^{IC}$ are identical, a behavioral analysis of ACCAs may be restricted to one of the quarters P^I of the pyramid P (Figure 3.4), where

$$P^I = \{(a,b,c) \in P \mid a + b + c \leq 1 \wedge a \geq b\} .$$

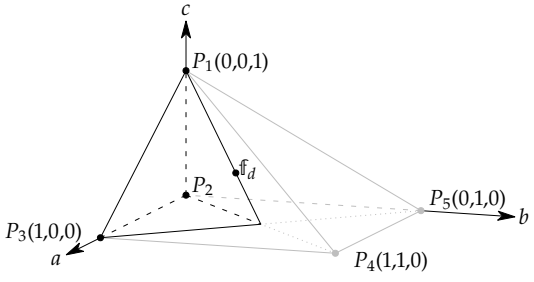


Figure 3.4: The pyramid P^I: parameterization of the reduced set of one-dimensional, density-conserving ACCAs with unit neighborhood radius.

3.5 Two-dimensional ACCAs

For the sake of simplicity, in the two-dimensional case we restrict to ACCAs with $r = 1$, *i.e.*, to two-dimensional ACCAs with the von Neumann neighborhood. To define the local rule $\mathbb{f}$ of a two-dimensional ACCA with the von Neumann neighborhood, we start by defining the values of $\mathbb{f}$ on $\{0,1\}^5$, in other words we fix the values $l_i \in [0,1]$. We can reuse Table 2.2 to represent the LUT of two-dimensional ACCAs, bearing in mind that the LUT entries are not only binary values.

There is a unique extension of $\mathbb{f}$ to the entire set $[0,1]^5$ that is affine in each variable. Namely, for each $x, y, z, u, v \in [0,1]$, we put

$$\mathbb{f}\left(y \begin{smallmatrix} x \\ z \\ v \end{smallmatrix} u\right) = \sum_{i=0}^{31} l_i \, h(x, i_4)\, h(y, i_3)\, h(z, i_2)\, h(u, i_1)\, h(v, i_0) \,, \tag{3.15}$$

where $i_4 i_3 i_2 i_1 i_0$ is the binary representation of the number i. Thus $\mathbb{f}$ can be expressed as the following polynomial:

$$\begin{aligned}\mathbb{f}\left(y \begin{smallmatrix} x \\ z \\ v \end{smallmatrix} u\right) =& l_0(1-x)(1-y)(1-z)(1-u)(1-v) \\ &+ l_1(1-x)(1-y)(1-z)(1-u)v \\ &+ l_2(1-x)(1-y)(1-z)u(1-v) \\ &\vdots \\ &+ l_{30}xyzu(1-v) \\ &+ l_{31}xyzuv \,.\end{aligned}$$

The canonical form of this polynomial is very long

$$\mathbb{f}\left(y \begin{smallmatrix} x \\ z \\ v \end{smallmatrix} u\right) = a_{31}xyzuv + a_{30}xyzu + a_{29}xyzv + \ldots + a_1 v + a_0 \,,$$

where the coefficients a_i are linear combinations of the l_i. For example

$$\begin{aligned}a_{30} =& l_{30} - l_{28} - l_{26} - l_{22} - l_{14} + l_{24} + l_{20} + l_{18} \\ &+ l_{12} + l_{10} + l_6 - l_{16} - l_8 - l_4 - l_2 + l_0 \,.\end{aligned}$$

3.6 Density-conserving two-dimensional ACCAs

In this section, we focus on a subset of a two-dimensional ACCAs, namely the density-conserving two-dimensional ACCAs. We start with theoretical definitions and later we will briefly analyze the structure of the ACCAs space.

3.6.1 Preliminaries

The next theorem gives a full characterization of two-dimensional density-conserving CCAs with the von Neumann neighborhood. Its proof can be found in [55] or it can be obtained from the formulas presented in [56].

Theorem 3.6.1 *A two-dimensional CCA with the von Neumann neighborhood is density-conserving if and only if, for all* $x, y, z, u, v \in [0,1]$*, its local rule* $\mathbb{f}$ *satisfies*

$$\begin{aligned}\mathbb{f}\begin{pmatrix}0&x&0\\y&z&u\\0&v&0\end{pmatrix} = x &+ \mathbb{f}\begin{pmatrix}0&0&0\\y&0&0\\0&v&0\end{pmatrix} - \mathbb{f}\begin{pmatrix}0&0&0\\x&0&0\\0&u&0\end{pmatrix} + \mathbb{f}\begin{pmatrix}0&0&0\\0&0&u\\0&v&0\end{pmatrix} - \mathbb{f}\begin{pmatrix}0&0&0\\0&0&x\\0&y&0\end{pmatrix}\\ &+ \mathbb{f}\begin{pmatrix}0&0&0\\0&z&u\\0&0&0\end{pmatrix} - \mathbb{f}\begin{pmatrix}0&0&0\\0&y&z\\0&0&0\end{pmatrix} + \mathbb{f}\begin{pmatrix}0&0&0\\0&z&0\\0&v&0\end{pmatrix} - \mathbb{f}\begin{pmatrix}0&0&0\\0&x&0\\0&z&0\end{pmatrix}\\ &+ \mathbb{f}\begin{pmatrix}0&0&0\\0&y&0\\0&0&0\end{pmatrix} - \mathbb{f}\begin{pmatrix}0&0&0\\0&z&0\\0&0&0\end{pmatrix} + \mathbb{f}\begin{pmatrix}0&0&0\\0&0&z\\0&0&0\end{pmatrix} - \mathbb{f}\begin{pmatrix}0&0&0\\0&0&u\\0&0&0\end{pmatrix}\\ &+ \mathbb{f}\begin{pmatrix}0&0&0\\0&0&0\\0&y&0\end{pmatrix} + \mathbb{f}\begin{pmatrix}0&0&0\\0&0&0\\0&z&0\end{pmatrix} + \mathbb{f}\begin{pmatrix}0&0&0\\0&0&0\\0&u&0\end{pmatrix} - \mathbb{f}\begin{pmatrix}0&0&0\\0&0&0\\0&x&0\end{pmatrix} - 2\mathbb{f}\begin{pmatrix}0&0&0\\0&0&0\\0&v&0\end{pmatrix}.\end{aligned} \tag{3.16}$$

Note that if $\mathbb{f}$ is affine in each variable, we can replace the condition "for all $x, y, z, u, v \in [0,1]$" by "for all $x, y, z, u, v \in \{0,1\}$", according to [49]. Consequently, to find necessary and sufficient conditions for two-dimensional ACCAs to be density-conserving, it suffices to solve a system of 32 linear algebraic equations involving the entries of the LUT. Doing this leads to Corollary 3.6.2.

Corollary 3.6.2 *A two-dimensional ACCA with the von Neumann neighborhood is density-conserving if and only if the entries of the LUT of its local rule satisfy*

$$\begin{cases}
l_0 = 0\,, & l_{16} = 1-a-b-d-g\,,\\
l_1 = a\,, & l_{17} = 1-b-d-g\,,\\
l_2 = b\,, & l_{18} = 1-d-h\,,\\
l_3 = c\,, & l_{19} = 1-b+c-d-h\,,\\
l_4 = d\,, & l_{20} = 1-b+d-e-g\,,\\
l_5 = e\,, & l_{21} = 1-b-g\,,\\
l_6 = f\,, & l_{22} = 1+a-b-e+f-h\,,\\
l_7 = -a-b+c-d+e+f\,, & l_{23} = 1-2b+c-d+f-h\,,\\
l_8 = g\,, & l_{24} = 1-c-d\,,\\
l_9 = h\,, & l_{25} = 1-c-d-g+h\,,\\
l_{10} = b+g\,, & l_{26} = 1+a+b-c-d-h+g\,,\\
l_{11} = -a+c+h\,, & l_{27} = 1-d\,,\\
l_{12} = b+2d-f+g\,, & l_{28} = 1+a+b-c+2d-e-f\,,\\
l_{13} = -a+b+d+e-f+h\,, & l_{29} = 1+b-c+d-f-g+h\,,\\
l_{14} = b+d+g\,, & l_{30} = 1+2a+b-c+d-e+g-h\,,\\
l_{15} = -2a+c+e+h\,, & l_{31} = 1\,,
\end{cases} \tag{3.17}$$

where the point (a, b, c, d, e, f, g, h) *belongs to an eight-dimensional polytope* $P \subset [0,1]^8$*, whose shape follows from the assumption that all* l_i *lie in the range* $[0,1]$.

As a consequence of Corollary 3.6.2, Eq. (3.15) simplifies to

$$\begin{aligned} \mathbb{f}\left(y \begin{smallmatrix} x \\ z \\ v \end{smallmatrix} u\right) =& (a+b-c)(xy-uv) + (a+d-e)(xz-zv) \\ &+ (a+g-h)(xu-yv) + (b+d-f)(yz-zu) \\ &+ (1-a-b-d-g)x + gy + dz + bu + av\ . \end{aligned} \tag{3.18}$$

In particular, for $(a,b,c,d,e,f,g,h) = \left(\frac{1}{5},\frac{1}{5},\frac{2}{5},\frac{1}{5},\frac{2}{5},\frac{2}{5},\frac{1}{5},\frac{2}{5}\right)$, we get the diffusion rule $\mathbb{f}_d$ from Eq. (2.8).

3.6.2 Structure of the space of density-conserving two-dimensional ACCAs

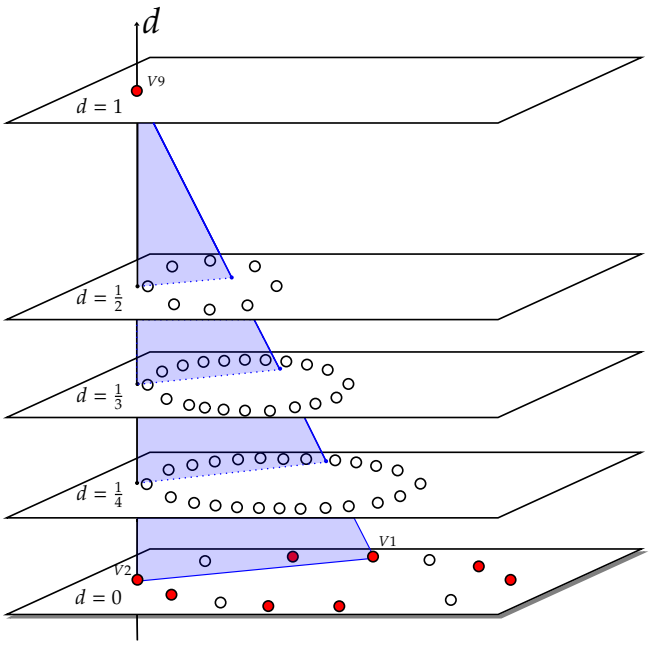


Figure 3.5: A scheme of the polytope P with a visualization of five seven-dimensional hyperplanes containing all 65 vertices of P. The vertices that correspond to binary CAs are marked in red, while the others are marked in white. The ACCAs whose local rule is independent of y and v are parameterized by points in the blue triangle.

Using the computer algebra system SageMath[1], we found that the polytope P has 65 vertices and 33 seven-dimensional facets. Let us recall that in the one-dimensional case, when the density-conserving ACCAs are parameterized by a three-dimensional pyramid (Figure 3.2), its five vertices correspond to well-known density-conserving ECAs: ECA204, ECA170, ECA240, ECA184 and ECA226 (see Table 2.3). As a consequence, we get that in one dimension each density-conserving ACCA is a convex combination of five density-conserving ECAs. Since there are only nine two-dimensional binary density-conserving CAs with the von Neumann neighborhood [57] (the identity CA and the shift and traffic CA in each of four directions), we see that the two-dimensional case is not analogous to one-dimensional counterpart. For example, we get a somewhat surprising fact that there are two-dimensional density-conserving ACCAs that are not a convex combination of two-dimensional density-conserving binary CAs.

All 65 vertices of the polytope P lie in five hyperplanes: 12 vertices are located in the seven-dimensional facet contained in the hyperplane $d = 0$ (we will call this facet "base"). The remaining vertices lie in four hyperplanes parallel to the base: $d = \frac{1}{4}$ (24 vertices), $d = \frac{1}{3}$ (20 vertices),

1: SageMath is a free and open-source computer algebra system initially released on 24th February 2005. It is an alternative for Mathematica and MATLAB. For our computation we used SageMath in Version 8.1.

$d = \frac{1}{2}$ (eight vertices) and $d = 1$ (one vertex – we will call this vertex "the top" of P) (see Figure 3.5).

3.6.3 Symmetries in the space of density-conserving two-dimensional ACCAs

We will investigate the structure of P and the behavior of density-conserving ACCAs by taking advantage of a few symmetries that are described in Section 2.7.

Let T be any transformation from the group of symmetries $\mathcal{T}$. If a local rule $\mathbb{f}$ is described by parameters a, b, c, d, e, f, g, h, then the local rule $\mathbb{f}^T$ is described by parameters $a^T, b^T, c^T, d^T, e^T, f^T, g^T, h^T$, where there are easy dependencies between these vectors of parameters. For example, the expression of the parameters $a^T, \ldots, h^T$ corresponding to the local rule $\mathbb{f}^T$ for the transformations I, R and C are given in the Table 3.1.

Table 3.1: Expression of the parameters $a^T, \ldots, h^T$ corresponding to the two-dimensional local rule $\mathbb{f}^T$ for the transformations I, R and C.

$T = I$	$T = R$	$T = C$
$a^T = a$	$a^T = g$	$a^T = -2a - b + c - d + e - g + h$
$b^T = g$	$b^T = a$	$b^T = -b + c - d + f + g - h$
$c^T = h$	$c^T = h$	$c^T = -a - b + c - 2d + e + f$
$d^T = d$	$d^T = d$	$d^T = d$
$e^T = e$	$e^T = b + 2d - f + g$	$e^T = -a - b + c + d - g + h$
$f^T = b + 2d - f + g$	$f^T = e$	$f^T = c + d + g - h$
$g^T = b$	$g^T = 1 - a - b - d - g$	$g^T = 2b - c + d - f + h$
$h^T = c$	$h^T = 1 - c - d$	$h^T = -a + b + e - f + h$

We see that $d^T = d$ for any $T \in \mathcal{T}$. As a consequence, we obtain that if $\pi_{\mathbb{f}}$ denotes the point in the polytope P representing the local rule $\mathbb{f}$, then for any $T \in \mathcal{T}$, the local rule $\mathbb{f}^T$ is represented by the point $\pi_{\mathbb{f}^T}$, lying in the same cross-section obtained by cutting the polytope P with a hyperplane parallel to its base.

We labeled each vertex of P using an index denoting the position of the vertex in the set of vertices, ordered in decreasing order by the number of adjacent vertices. As a result we get an ordered set of vertices $V1, \ldots, V65$, where $V1$ has the highest number of adjacent vertices. The vertices correspond to the local rules $\mathbb{f}_{V1}, \mathbb{f}_{V2}, \ldots, \mathbb{f}_{V65}$, among which there are only 10 independent ones represented by a vertex with the lowest index in a given group: $\mathbb{f}_{V1}, \mathbb{f}_{V5}, \mathbb{f}_{V9}, \mathbb{f}_{V10}, \mathbb{f}_{V18}, \mathbb{f}_{V22}, \mathbb{f}_{V26}, \mathbb{f}_{V34}, \mathbb{f}_{V42}, \mathbb{f}_{V46}$, while the others are their transformations. These vertices are presented in Table 3.2.

Using the vertex labeling introduced in Table 3.2, we can describe the position of the points related to binary ACCAs:

- $V9$ (the top of the polytope P) – the identity CA,
- $V1, V2, V3, V4$ – the shift CAs (left, down, up and right, respectively),

Table 3.2: The labeling of the vertices of the polytope P. The transformations considered are: Id – identity, H, I – horizontal and vertical axial flips along perpendicular bisectors ($H = IR^2$), A_1, A_2 – axial flips along diagonal bisectors ($A_1 = RI, A_2 = IR$), R, R^2, R^3 - rotation by 90, 180 and 270 degrees), C – conjugation.

V	a	b	c	d	e	f	g	h	Id	H	I	A_1	A_2	R	R^2	R^3	C	CH	CI	CA_1	CA_2	CR	CR^2	CR^3
1	0	1	1	0	0	1	0	0	1	1	4	3	2	2	4	3	1	1	4	3	2	2	4	3
5	0	0	0	0	1	0	0	0	5	8	5	7	6	7	8	6	8	5	8	6	7	6	5	7
9	0	0	0	1	1	1	0	0	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9
10	0	$\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{2}$	0	1	0	$\frac{1}{2}$	10	15	12	17	11	16	14	13	14	12	15	11	17	13	10	16
18	0	0	$\frac{1}{2}$	0	0	0	0	$\frac{1}{2}$	18	21	18	19	20	19	21	20	21	18	21	20	19	20	18	19
22	0	0	$\frac{2}{3}$	$\frac{1}{3}$	0	0	$\frac{1}{3}$	$\frac{1}{3}$	22	25	24	22	23	25	23	24	23	24	25	23	22	24	22	25
26	$\frac{1}{3}$	$\frac{1}{3}$	$\frac{2}{3}$	$\frac{1}{3}$	$\frac{1}{3}$	1	0	$\frac{2}{3}$	26	33	28	30	29	31	32	27	30	31	27	26	32	33	29	28
34	0	$\frac{1}{2}$	0	$\frac{1}{4}$	$\frac{3}{4}$	1	0	$\frac{1}{4}$	34	41	38	36	35	40	37	39	37	38	41	35	36	39	34	40
42	0	$\frac{1}{3}$	$\frac{2}{3}$	$\frac{1}{3}$	0	$\frac{2}{3}$	0	$\frac{1}{3}$	42	60	58	45	43	62	44	59	60	42	44	59	62	43	58	45
46	0	$\frac{1}{4}$	$\frac{3}{4}$	$\frac{1}{4}$	0	$\frac{1}{4}$	0	$\frac{1}{4}$	46	63	64	56	48	61	54	65	49	53	47	57	50	55	52	51

- $V5, V6, V7, V8$ – the traffic CAs (down, left, right and up, respectively).

From Table 3.2 we conclude that the base of the polytope P contains the following 12 vertices: eight corresponding to binary CAs (shift and traffic) and four additional ones ($V18, V19, V20, V21$). To understand their action, below we consider the local rule $\mathbb{f}_{V18}$ (the other three are transformations of this rule):

$$\begin{aligned}\mathbb{f}_{V18}\left(\begin{smallmatrix} y & x \\ z & u \\ v & \end{smallmatrix}\right) &= -\frac{1}{2}(xy - uv) - \frac{1}{2}(xu - yv) + x \\ &= \left(1 - \frac{y+u}{2}\right)x + \frac{y+u}{2}v\ .\end{aligned}$$

So, the local rule $\mathbb{f}_{V18}$ acts as a weighted average of x and v with weights that depend on the sum $y + u$, which makes this rule an exemplary one for those that are not a convex combination of binary CAs.

3.6.4 Density-conserving two-dimensional ACCAs with independent y and v

There is a kind of density-conserving ACCAs having a local rule that is independent of y and v (*i.e.*, it depends only on the upper right corner of the neighborhood). The general form of such a local rule is the following:

$$\mathbb{f}\left(\begin{smallmatrix} y & x \\ z & u \\ v & \end{smallmatrix}\right) = (1 - b - d)x + dz + bu\ , \tag{3.19}$$

where $b, d \in [0, 1]$ and $b + d \leq 1$. The corresponding points of the polytope P lie in the triangle with vertices $V1, V2$ and $V9$.

3.7 Rotation-symmetric two-dimensional ACCAs

As two-dimensional CAs that can be used to model physical, biological and chemical phenomena are often invariant under rotation of the neighborhood by 90 degrees, we dedicate part of our attention to this

kind of ACCAs. Let us recall the definition of a two-dimensional rotation-symmetric local rule $\mathbb{f}$.

Definition 3.7.1 *Let $\mathbb{f}$ be the local rule of a two-dimensional CA. We say that the rule $\mathbb{f}$ is rotation-symmetric if it satisfies:*

$$\mathbb{f}\left(y \begin{smallmatrix} x \\ z \\ v \end{smallmatrix} u\right) = \mathbb{f}\left(v \begin{smallmatrix} y \\ z \\ u \end{smallmatrix} x\right) , \tag{3.20}$$

for all x, y, z, u, v from the considered state set.

In other words, $\mathbb{f}$ is rotation-symmetric if $\mathbb{f}^R = \mathbb{f}$. According to [49], an ACCA is rotation-symmetric if and only if Eq. (3.20) is satisfied for any $x, y, z, u, v \in \{0, 1\}$. Thus, we have the following characterization of rotation-symmetric ACCAs.

Theorem 3.7.1 *A two-dimensional ACCA with the von Neumann neighborhood is rotation-symmetric if the entries of the LUT of its local rule satisfy*

$$\begin{cases} l_1 = l_2 = l_8 = l_{16} , \\ l_3 = l_9 = l_{18} = l_{24} , \\ l_5 = l_6 = l_{12} = l_{20} , \\ l_7 = l_{13} = l_{22} = l_{28} , \\ l_{10} = l_{17} , \\ l_{11} = l_{19} = l_{25} = l_{26} , \\ l_{14} = l_{21} , \\ l_{15} = l_{23} = l_{29} = l_{30} . \end{cases} \tag{3.21}$$

Thus, if we identify the set of all ACCAs with $[0, 1]^{32}$, then the set of rotation-symmetric ACCAs is a 12-dimensional subset of it.

Since an outer-totalistic CA is a CA whose local rule depends only on the state of the central cell (i, j) and the sum of the states of the cells adjacent to (i, j), thus obviously, if some CA is outer-totalistic, then it is rotation-symmetric. But one can see that if for some ACCA it holds that $l_3 \neq l_{10}$ or $l_7 \neq l_{14}$, then this ACCA is definitely not outer-totalistic. This means in particular that the set of outer-totalistic ACCAs is a proper subset of the set of rotation-symmetric ACCAs. However, it turns out that if we focus on density-conserving ACCAs, then being rotation-symmetric is equivalent to being outer-totalistic, as a combination of Eq. (3.17) and Eq. (3.21) gives the following form of a local rule:

$$\mathbb{f}\left(y \begin{smallmatrix} x \\ z \\ v \end{smallmatrix} u\right) = \alpha x + \alpha y + (1 - 4\alpha)z + \alpha u + \alpha v , \tag{3.22}$$

where $\alpha \in \left[0, \frac{1}{4}\right]$. Note, that for $\alpha = \frac{1}{5}$ we get the diffusion rule defined in Eq. (2.8). The density-conserving and rotation-symmetric ACCA rules are visualized in Figure 3.6 and the general form of their LUT is presented in Table 3.3. The rotation-symmetric density-conserving ACCAs correspond to the points of P lying on the line segment connecting the center of the base with the top $V9$.

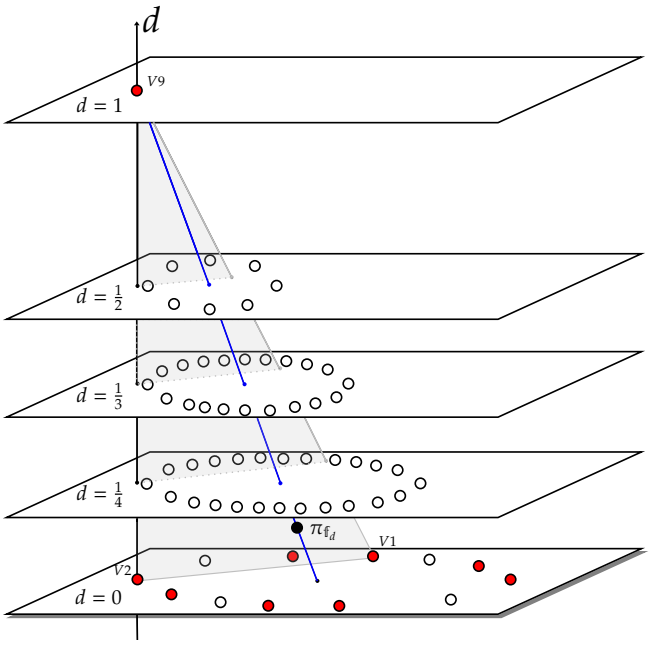


Figure 3.6: A scheme of the polytope P with a visualization of five seven-dimensional hyperplanes containing all 65 vertices of P. The vertices that correspond to binary CAs are marked in red, while the others are marked in white. The blue line segment consists of the points corresponding to rotation-symmetric ACCAs. The black point π_{f_d} corresponds to the diffusion ACCA. The ACCAs whose local rule is independent of y and v are parameterized by points in the greyed-out triangle.

Table 3.3: LUT of a two-dimensional, density-conserving and rotation-symmetric ACCA rule with the von Neumann neighborhood parameterized by $\alpha \in [0, \frac{1}{4}]$.

l_0: 0	l_1: α	l_2: α	l_3: 2α
l_4: $1-4\alpha$	l_5: $1-3\alpha$	l_6: $1-3\alpha$	l_7: $1-2\alpha$
l_8: α	l_9: 2α	l_{10}: 2α	l_{11}: 3α
l_{12}: $1-3\alpha$	l_{13}: $1-2\alpha$	l_{14}: $1-2\alpha$	l_{15}: $1-\alpha$
l_{16}: $1-4\alpha$	l_{17}: 2α	l_{18}: 2α	l_{19}: 3α
l_{20}: $1-3\alpha$	l_{21}: $1-2\alpha$	l_{22}: $1-2\alpha$	l_{23}: $1-\alpha$
l_{24}: 2α	l_{25}: 3α	l_{26}: 3α	l_{27}: 4α
l_{28}: $1-2\alpha$	l_{29}: $1-\alpha$	l_{30}: $1-\alpha$	l_{31}: 1

3.8 Conclusions

In this chapter, we defined ACCAs, an extension of CCAs for one and two dimensions and defined a few groups of ACCAs. We see that the set of two-dimensional ACCAs is huge, but contains smaller subsets with an interesting behavior. In the next chapter, we will define the focal problem of our work where ACCAs will have a major role in the solution.

The Density Classification Problem

4

In this chapter, we formulate the Density Classification Problem (DCP). We start with a historical background and later identify two groups of the DCP in the context of ACCA.

4.1 Preliminaries 33
4.2 Classical DCP formulation 34
4.3 Relaxed DCP formulation . 36
4.4 Conclusions 40

4.1 Preliminaries

After its introduction by Gács *et al.* [58], the density classification problem was investigated more elaborately by Packard [59]. Essentially, this problem boils down to a quest for a CA that is able to determine whether the total number of 1s in its initial configuration is greater than the total number of 0s, in such a way that it evolves to a homogeneous configuration of 1s, and 0s otherwise. This process is represented in Figure 4.1.

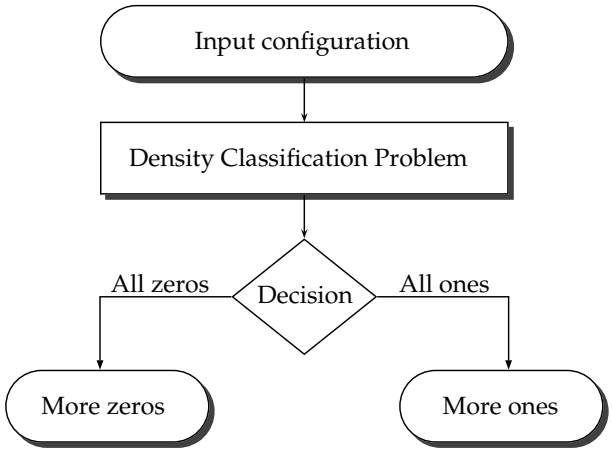


Figure 4.1: Flow diagram of the Density Classification Problem.

In 1995 it was proved that for any dimension d there exists no d-dimensional binary CA that solves the DCP [60]. The proof follows from the fact that one of the necessary conditions for a binary CA to classify density (in any sense) for all initial configurations (of any size) is that it must be density-conserving [61], conflicting with the goal of the DCP.

Yet, certain CAs are capable of solving this problem for many initial configurations. One of the best-known ones is the Gacs–Kurdyumov–Levin (GKL) rule [58], which leads to a correct solution for more than 80% of the initial configurations, while similar CAs were found using genetic algorithms [62, 63]. In order to obtain an infallible classifier, many scholars modified the original problem by allowing two CAs (instead of one) [64], embedding memory [65], or using probabilistic CAs [66–68]. In other studies, the output specification of the problem is changed [69] by accepting heterogeneous configurations that obey certain properties as valid answers. Yet, this implies that multiple time steps (and memory) are needed to read the answer, or that the states of multiple cells (global access) are accessed.

The search for an infallible classifier applicable to all initial configurations has also given rise to an approach that involves CCAs. For example, Briceño *et al.* [70] propose a large diffusion and small amplification CA achieving a correct solution of the DCP for any number of cells. Yet, it comes along with a parameter based on the number of cells, thus one ends up with a family of CCAs rather than a single CCA.

In the next section, we present the classical formulation of the DCP for CCAs and the reasons why there is no ACCA that is able to solve it. Subsequently, we formulate a relaxed version of the DCP.

4.2 Classical formulation of the DCP

In this section, we recall the classical formulation of the DCP and show that there exists no solution in the set of ACCAs.

Definition 4.2.1 (Density classification problem in the classical sense) *Let $\rho_0 \in (0,1)$ and $\mathbb{F}$ be the global rule of a CCA. We say that this CCA* solves the DCP at threshold ρ_0 in the classical sense *if for any $\boldsymbol{x} \in \{0,1\}^\star$, it holds that:*

$$\begin{aligned} \rho(\boldsymbol{x}) < \rho_0 \quad &\Rightarrow \quad \lim_{t\to\infty} \mathbb{F}^t(\boldsymbol{x}) = (0,0,\ldots,0)\,, \\ \rho(\boldsymbol{x}) > \rho_0 \quad &\Rightarrow \quad \lim_{t\to\infty} \mathbb{F}^t(\boldsymbol{x}) = (1,1,\ldots,1)\,. \end{aligned}$$

The above condition means that the orbit of any initial configuration $\boldsymbol{x} \in \{0,1\}^n$ converges to $(0,0,\ldots,0)$ or $(1,1,\ldots,1)$, depending on the number of 1s in $\boldsymbol{x}$. Clearly, a CCA that is density-conserving cannot solve the classical DCP at any threshold.

Definition 4.2.2 (Threshold-conserving CCA) *Let $\rho_0 \in (0,1)$ and $\mathbb{F}$ be the global rule of a CCA. We say that this CCA* conserves the threshold ρ_0 *if for any $\boldsymbol{x} \in \{0,1\}^*$, it holds that:*

$$\begin{aligned} \rho(\boldsymbol{x}) < \rho_0 \quad &\Rightarrow \quad \rho\left(\mathbb{F}(\boldsymbol{x})\right) \le \rho_0\,, \\ \rho(\boldsymbol{x}) > \rho_0 \quad &\Rightarrow \quad \rho\left(\mathbb{F}(\boldsymbol{x})\right) \ge \rho_0\,. \end{aligned}$$

In the above definitions, there is no condition if $\rho(\boldsymbol{x}) = \rho_0$, which is in agreement with the formulation in [71]. Obviously, every density-conserving CCA is also a threshold-conserving CCA, irrespective of the threshold ρ_0.

Theorem 4.2.1 *Let $\rho_0 \in (0,1)$. There exists no ACCA that solves the DCP at threshold ρ_0 in the classical sense and that conserves the threshold ρ_0.*

The proof of this theorem relies on Lemmas 4.2.2 and 4.2.3 below. They can be proved in a similar way as the corresponding facts for CAs in [60, 71, 72], but we give a detailed explanation for the reader's convenience.

Lemma 4.2.2 *Let $\mathbb{f}$ be the local rule of a CCA that solves the DCP at threshold $\rho_0 \in (0,1)$ in the classical sense and let $\mathbb{F}$ be the corresponding global rule. If the function $\mathbb{f}$ is continuous, then $\mathbb{f}(0,0,\dots,0) = 0$ and $\mathbb{f}(1,1,\dots,1) = 1$.*

Proof. Let us define the function $g\colon [0,1] \to [0,1]$ so that $g(x) = \mathbb{f}(x,x,\dots,x)$. We need to show that $g(0) = 0$ and $g(1) = 1$. Let us define the sequence $c_0 = 0$ and $c_{t+1} = g(c_t)$ for $t \geq 0$. Then, for any t it holds that:

$$\mathbb{F}^t(0,0,\dots,0) = (c_t, c_t, \dots, c_t)\ .$$

Since $\mathbb{F}^t(0,0,\dots,0) \overset{t\to\infty}{\to} (0,0,\dots,0)$, we know that $c_t \overset{t\to\infty}{\to} 0$. Using the continuity of g, we get $g(c_t) \overset{t\to\infty}{\to} g(0)$, and as a result we have:

$$g(0) = \lim_{t\to\infty} g(c_t) = \lim_{t\to\infty} c_{t+1} = \lim_{t\to\infty} c_t = 0\ .$$

The proof of $g(1) = 1$ is analogous. □

Lemma 4.2.3 *Let $\mathbb{f}$ be the local rule of a CCA that conserves the threshold $\rho_0 \in (0,1)$ and let $\mathbb{F}$ be the corresponding global rule. If $\mathbb{f}(0,0,\dots,0) = 0$ and $\mathbb{f}(1,1,\dots,1) = 1$, then for any $\boldsymbol{x} \in \{0,1\}^*$, it holds that $\rho\left(\mathbb{F}(\boldsymbol{x})\right) = \rho(\boldsymbol{x})$.*

Proof. Suppose that the assertion of this lemma is false. Then there exist an $n \in \mathbb{N}$ and $\boldsymbol{x} \in \{0,1\}^n$ such that $\rho\left(\mathbb{F}(\boldsymbol{x})\right) \neq \rho(\boldsymbol{x})$. We assume that $n \geq r$, since otherwise we can replace $\boldsymbol{x}$ by $(\boldsymbol{x})^r$, because for each $j \in \mathbb{N}$ we have that $\rho\left((\boldsymbol{x})^j\right) = \rho(\boldsymbol{x})$ and $\mathbb{F}\left((\boldsymbol{x})^j\right) = (\mathbb{F}(\boldsymbol{x}))^j$, as a consequence of the periodic boundary conditions.

We choose $L \in \mathbb{N}$ such that $|\sigma(\boldsymbol{x}) - \sigma(\mathbb{F}(\boldsymbol{x}))| > \frac{4r+2}{L}$. Let $\hat{\boldsymbol{x}}$ denote the first r bits and $\check{\boldsymbol{x}}$ the last r bits of configuration $\boldsymbol{x}$, and choose $m, k \in \mathbb{N}$ greater than $2r+1$ such that:

$$\frac{\sigma(\check{\boldsymbol{x}}\hat{\boldsymbol{x}}) + L\sigma(\boldsymbol{x}) + k - 1}{nL + 2r + m + k} < \rho_0 < \frac{\sigma(\check{\boldsymbol{x}}\hat{\boldsymbol{x}}) + L\sigma(\boldsymbol{x}) + k + 1}{nL + 2r + m + k}\ .$$

Note that this choice is always possible, as for every $j \in \mathbb{N}$ we have:

$$\lfloor 10^j\rho_0 \rfloor - 1 < 10^j\rho_0 < \lfloor 10^j\rho_0 \rfloor + 1\ ,$$

and both $\lfloor 10^j\rho_0 \rfloor$ and $10^j - \lfloor 10^j\rho_0 \rfloor$ approach infinity as j grows. Now, let us consider the following two configurations:

$$\boldsymbol{u} = (\boldsymbol{x})^L\hat{\boldsymbol{x}}(1)^{k-1}(0)^{m+1}\check{\boldsymbol{x}}\ ,$$
$$\boldsymbol{v} = (\boldsymbol{x})^L\hat{\boldsymbol{x}}(1)^{k+1}(0)^{m-1}\check{\boldsymbol{x}}\ .$$

We see that:

$$\mathbb{F}(\boldsymbol{u}) = \mathbb{F}\left((\boldsymbol{x})^L\hat{\boldsymbol{x}}(1)^{k-1}(0)^{m+1}\check{\boldsymbol{x}}\right) = (\mathbb{F}(\boldsymbol{x}))^L\, A(1)^{k-1-2r}B(0)^{m+1-2r}C\ ,$$
$$\mathbb{F}(\boldsymbol{v}) = (\mathbb{F}(\boldsymbol{x}))^L A(1)^{k+1-2r}B(0)^{m-1-2r}C\ ,$$

where the strings A, B, C have length $2r$. According to the choice of m and k, it holds that

$$\rho(\boldsymbol{u}) < \rho_0 < \rho(\boldsymbol{v})\ ,$$

which results in

$$\rho(\mathbb{F}(\boldsymbol{u})) \leq \rho_0 \leq \rho(\mathbb{F}(\boldsymbol{v}))\ .$$

We conclude that

$$\sigma(\mathbb{F}(\boldsymbol{v})) - \sigma(\boldsymbol{u}) > 0 \text{ and } \sigma(\mathbb{F}(\boldsymbol{u})) - \sigma(\boldsymbol{v}) < 0\ .$$

An easy computation shows that

$$\sigma(\mathbb{F}(\boldsymbol{v})) - \sigma(\boldsymbol{u}) = L\,(\sigma(\mathbb{F}(\boldsymbol{x})) - \sigma(\boldsymbol{x})) + \sigma(ABC) - \sigma(\check{\boldsymbol{x}}\hat{\boldsymbol{x}}) - 2\,r + 2\ ,$$

while

$$\sigma(\mathbb{F}(\boldsymbol{u})) - \sigma(\boldsymbol{v}) = L\,(\sigma(\mathbb{F}(\boldsymbol{x})) - \sigma(\boldsymbol{x})) + \sigma(ABC) - \sigma(\check{\boldsymbol{x}}\hat{\boldsymbol{x}}) - 2\,r - 2\ .$$

So

$$L\,(\sigma(\mathbb{F}(\boldsymbol{x})) - \sigma(\boldsymbol{x})) > \sigma(\check{\boldsymbol{x}}\hat{\boldsymbol{x}}) - \sigma(ABC) + 2r - 2 \geq -4\,r - 2\ ,$$

while

$$L\,(\sigma(\mathbb{F}(\boldsymbol{x})) - \sigma(\boldsymbol{x})) < \sigma(\check{\boldsymbol{x}}\hat{\boldsymbol{x}}) - \sigma(ABC) + 2r + 2 \leq 4\,r + 2\ .$$

Thus, we have shown that

$$L\,|\sigma(\mathbb{F}(\boldsymbol{x})) - \sigma(\boldsymbol{x})| < 4\,r + 2\ ,$$

contradicting our earlier choice of L. □

Proof of Theorem 4.2.1. Let $\rho_0 \in (0,1)$ be given. Suppose that there is an ACCA that solves the classical DCP at the threshold ρ_0, and that this ACCA conserves the same threshold. Let $\mathbb{f}$ and $\mathbb{F}$ be its local and global rule, respectively. Due to Lemma 4.2.2 we have that $l_0 = 0$ and $l_s = 1$, which implies that the assumptions of Lemma 4.2.3 are satisfied, whence $\rho\,(F(\boldsymbol{x})) = \rho(\boldsymbol{x})$ for any $\boldsymbol{x} \in \{0,1\}^*$. Since $\mathbb{f}$ is affine in each variable, it then follows that $\rho\,(F(\boldsymbol{x})) = \rho(\boldsymbol{x})$ for any $\boldsymbol{x} \in [0,1]^*$ according to [49]. Hence, the ACCA defined by the local rule $\mathbb{f}$ is density-conserving, which contradicts our conclusion based on Definition 4.2.1, as density-conserving CCAs cannot solve the DCP in the classical sense. □

4.3 Relaxed formulation of the DCP for ACCAs

In this section, we propose a relaxed formulation of the DCP that involves a modification of the output specification.

Definition 4.3.1 (Relaxed formulation of the density classification problem) *Let $\rho_0 \in (0,1)$ and $\mathbb{F}$ be the global rule of an ACCA. We say that this ACCA solves the DCP for any cellular space $\mathbf{C}$ at the threshold ρ_0 in the* relaxed sense *if it holds that:*

$$(\exists T)\left(\forall \boldsymbol{x} \in \{0,1\}^{\mathbf{C}}\right)(\forall t \geq T)\left(\begin{cases}\rho(\boldsymbol{x}) < \rho_0 \Rightarrow \mathbb{F}^t(\boldsymbol{x}) \in [0,\rho_0)^{\mathbf{C}} \\ \rho(\boldsymbol{x}) > \rho_0 \Rightarrow \mathbb{F}^t(\boldsymbol{x}) \in (\rho_0,1]^{\mathbf{C}}\end{cases}\right). \tag{4.1}$$

So, if the density of an initial configuration $\boldsymbol{x}$ is smaller (resp. greater) than ρ_0, then all values $x^t(i)$ are smaller (resp. greater) than ρ_0, for t large enough. For a given ACCA, threshold ρ_0 and n, the smallest T for which the condition in Eq. (4.1) is fulfilled will be called the *convergence time*, denoted by $T(n)$. It turns out that in the case of one dimension, we can simplify the above condition by means of the following lemma.

Lemma 4.3.1 *Let $\mathbb{F}$ be the global rule of a one-dimensional density-conserving ACCA with unit neighborhood radius. Then, for any $v \in [0,1]$, it holds that:*

$$\boldsymbol{x} \in [0,v]^n \Rightarrow \mathbb{F}(\boldsymbol{x}) \in [0,v]^n \text{ and } \boldsymbol{x} \in [v,1]^n \Rightarrow \mathbb{F}(\boldsymbol{x}) \in [v,1]^n \ .$$

Proof. Let $\mathbb{f}$ be the local rule corresponding to the global rule $\mathbb{F}$. Let $\boldsymbol{x} \in [0,v]^n$ for some $v \in [0,1]$. According to Theorem 3.1.1, to prove that $\mathbb{F}(\boldsymbol{x}) \in [0,v]^n$, it is sufficient to show that all eight values:

$$\begin{gathered}\mathbb{f}(0,0,0)\ ,\ \mathbb{f}(v,0,0)\ ,\ \mathbb{f}(0,v,0)\ ,\ \mathbb{f}(0,0,v)\ ,\\ \mathbb{f}(v,v,0)\ ,\ \mathbb{f}(v,0,v)\ ,\ \mathbb{f}(0,v,v)\ ,\ \mathbb{f}(v,v,v)\ ,\end{gathered}$$

are not greater than v.

Since $\mathbb{f}$ is given by Eq. (3.13), we have

$$\begin{aligned}\mathbb{f}(0,0,0) &= 0,\\ \mathbb{f}(v,0,0) &= l_4 v,\\ \mathbb{f}(0,v,0) &= l_2 v,\\ \mathbb{f}(0,0,v) &= l_1 v\ ,\end{aligned}$$

thus all of them are at most v, since $l_1, l_2, l_4 \in [0,1]$. Moreover, since $l_1 + l_2 + l_4 = 1$ (see Eq. (3.11)), then

$$\begin{aligned}\mathbb{f}(v,v,0) &= (l_6 + l_1 - 1)v^2 + (l_4 + l_2)v = v((l_6 + l_1 - 1)v + (1 - l_1))\\ &= v((1-v)(1-l_1) + v l_6)\ .\end{aligned}$$

Note that the expression $(1-v)(1-l_1+vl_6)$ is a weighted mean of the values $1-l_1$ and l_6, both not greater than 1, so it is also not greater than 1. Hence

$$\mathbb{f}(v,v,0) \le v\ .$$

Again using $l_1 + l_2 + l_4 = 1$, we get

$$\mathbb{f}(v,0,v) = (l_4 + l_1)v = (1 - l_2)v \le v\ .$$

According to Eq. (3.11)

$$l_6 + l_1 - 1 = 2 - l_3 - l_5 + l_1 - 1 = 1 - l_3 - (1 - l_2) + l_1 = l_1 + l_2 - l_3\ ,$$

thus

$$\begin{aligned}\mathbb{f}(0,v,v) &= -(l_6 + l_1 - 1)v^2 + (l_1 + l_2)v = v((l_1 + l_2) - (l_1 + l_2 - l_3)v)\\ &= v((1-v)(l_1 + l_2) + v l_3)\ .\end{aligned}$$

As $(1-v)(l_1 + l_2) + v l_3$ is a weighted mean of $l_1 + l_2$ and l_3 (and $l_1 + l_2 =$

$1 - l_4 \leq 1$), it is not greater than 1. Thus

$$\mathbb{f}(0, v, v) \leq v \ .$$

Finally, from Eq. (3.13)

$$\mathbb{f}(v, v, v) = v \ ,$$

which finishes the proof of the first part of the claim. To prove the second part, let us note that using similar arguments as above, we get

$$\begin{aligned}
&\mathbb{f}(v, v, v) = v \ , \\
&\mathbb{f}(v, v, 1) = v + (1 - v)((1 - v)l_1 + v(1 - l_6)) \geq v \ , \\
&\mathbb{f}(v, 1, v) = v + (1 - v)l_2 \geq v \ , \\
&\mathbb{f}(1, v, v) = v + (1 - v)((1 - v)l_4 + v(1 - l_3)) \geq v \ , \\
&\mathbb{f}(1, 1, v) = v + (1 - v)l_6 \geq v \ , \\
&\mathbb{f}(1, v, 1) = v + (1 - v)(1 - l_2) \geq v \ , \\
&\mathbb{f}(v, 1, 1) = v + (1 - v)((1 - v)(1 - l_4) + vl_3) \geq v \ , \\
&\mathbb{f}(1, 1, 1) = 1 \geq v \ ,
\end{aligned}$$

which means that $\boldsymbol{x} \in [v, 1]^n$ implies $\mathbb{F}(\boldsymbol{x}) \in [v, 1]^n$, according to Theorem 3.1.1. □

Note that Lemma 4.3.1 cannot be generalized to ACCAs that are not density-conserving. Following Lemma 4.3.1, we may rewrite Eq. (4.1) as:

$$(\forall n \in \mathbb{N})\,(\exists T)\,(\forall \boldsymbol{x} \in \{0,1\}^n)\left(\begin{cases} \rho(\boldsymbol{x}) < \rho_0 \Rightarrow \mathbb{F}^T(\boldsymbol{x}) \in [0, \rho_0)^n \\ \rho(\boldsymbol{x}) > \rho_0 \Rightarrow \mathbb{F}^T(\boldsymbol{x}) \in (\rho_0, 1]^n \end{cases}\right) . \tag{4.2}$$

Indeed, the conditions in Eqs. (4.1) and (4.2) are equivalent, since we know from Lemma 4.3.1 that if for some T it holds that $\mathbb{F}^T(\boldsymbol{x}) \in [0, \rho_0)^n$, then $\mathbb{F}^t(\boldsymbol{x}) \in [0, \rho_0)^n$ for every $t \geq T$, and similarly if $\mathbb{F}^T(\boldsymbol{x}) \in (\rho_0, 1]^n$, then $\mathbb{F}^t(\boldsymbol{x}) \in (\rho_0, 1]^n$ for every $t \geq T$.

In the two-dimensional case, the conditions in Eq. (4.1) can be also simplified, but the proof is much more difficult.

Theorem 4.3.2 *Let $\mathbb{F}$ be the global rule of a two-dimensional density-conserving ACCA with the von Neumann neighborhood. Then, for any $\varepsilon \in [0, 1]$, it holds that:*

$$\boldsymbol{x} \in [0, \varepsilon]^{\mathcal{C}} \Rightarrow \mathbb{F}(\boldsymbol{x}) \in [0, \varepsilon]^{\mathcal{C}} \text{ and } \boldsymbol{x} \in [\varepsilon, 1]^{\mathcal{C}} \Rightarrow \mathbb{F}(\boldsymbol{x}) \in [\varepsilon, 1]^{\mathcal{C}} \ .$$

Proof. Let $\mathbb{f}$ be the local rule corresponding to $\mathbb{F}$. Since the ACCA is density-conserving, the entries of the LUT of $\mathbb{f}$ satisfy Eq. (3.17). We start with the first part of our claim. According to Theorem 3.1.1, it suffices to show that for any $\boldsymbol{z} \in \{0, \varepsilon\}^5$, it holds that $\mathbb{f}(\boldsymbol{z}) \leq \varepsilon$. However, due to the symmetry of the von Neumann neighborhood, we do not need to consider all $\boldsymbol{z} \in \{0, \varepsilon\}^5$, but only some of them. First of all, we have

$$\mathbb{f}\begin{pmatrix} & 0 & \\ 0 & 0 & 0 \\ & 0 & \end{pmatrix} = 0 \text{ and } \mathbb{f}\begin{pmatrix} & \varepsilon & \\ \varepsilon & \varepsilon & \varepsilon \\ & \varepsilon & \end{pmatrix} = \varepsilon \ .$$

Since $a, d, \varepsilon \in [0,1]$, it holds that

$$\mathbb{f}\begin{pmatrix} 0 \\ 0\,0\,0 \\ \varepsilon \end{pmatrix} = a\varepsilon \leq \varepsilon \text{ and } \mathbb{f}\begin{pmatrix} 0 \\ 0\,\varepsilon\,0 \\ 0 \end{pmatrix} = d\varepsilon \leq \varepsilon\ ,$$

which covers all cases when $\boldsymbol{z} \in \{0, \varepsilon\}^5$ has one ε among its coordinates.

As $a, b, c, d \geq 0$, it holds that

$$\begin{aligned}\mathbb{f}\begin{pmatrix} \varepsilon \\ \varepsilon\,0\,0 \\ 0 \end{pmatrix} &= (a+b-c)\varepsilon^2 + (1-a-b-d)\varepsilon \\ &= \varepsilon - [(1-\varepsilon)(a+b) - \varepsilon c - d]\,\varepsilon \leq \varepsilon\ ,\end{aligned}$$

and, since $l_{10} = b + g \leq 1$,

$$\mathbb{f}\begin{pmatrix} 0 \\ \varepsilon\,0\,\varepsilon \\ 0 \end{pmatrix} = (b+g)\varepsilon \leq \varepsilon\ .$$

Moreover, as $l_{20} = 1 - b + d - e - g \leq 1$, it holds that $d - e \leq b + g$, and hence

$$\begin{aligned}\mathbb{f}\begin{pmatrix} \varepsilon \\ 0\,\varepsilon\,0 \\ 0 \end{pmatrix} &= (a+d-e)\varepsilon^2 + (1-a-b-g)\varepsilon \\ &\leq (a+b+g)\varepsilon^2 + (1-a-b-g)\varepsilon \\ &= \varepsilon - (1-\varepsilon)\varepsilon(a+b+g) \leq \varepsilon\ ,\end{aligned}$$

and this covers all cases when $\boldsymbol{z} \in \{0, \varepsilon\}^5$ has two ε's among its coordinates.

As $l_{28} \leq 1$, it holds that $a + b - c + 2d - e - f \leq 0$, and hence

$$\begin{aligned}\mathbb{f}\begin{pmatrix} \varepsilon \\ \varepsilon\,\varepsilon\,0 \\ 0 \end{pmatrix} &= (a+b-c)\varepsilon^2 + (a+d-e)\varepsilon^2 + (b+d-f)\varepsilon^2 + (1-a-b)\varepsilon \\ &= \varepsilon - (1-\varepsilon)\varepsilon(a+b) + (a+b-c+2d-e-f)\varepsilon^2 \leq \varepsilon\ .\end{aligned}$$

Since $l_{14} = b + d + g \leq 1$, it holds that

$$\mathbb{f}\begin{pmatrix} 0 \\ \varepsilon\,\varepsilon\,\varepsilon \\ 0 \end{pmatrix} = (b+d+g)\varepsilon \leq \varepsilon\ ,$$

while $l_{26} \leq 1$ implies that $a + b - c - d - h + g \leq 0$, thus

$$\begin{aligned}\mathbb{f}\begin{pmatrix} \varepsilon \\ \varepsilon\,0\,\varepsilon \\ 0 \end{pmatrix} &= (a+b-c)\varepsilon^2 + (a+g-h)\varepsilon^2 + (1-a-d)\varepsilon \\ &= \varepsilon - (1-\varepsilon)\varepsilon(a+d) + (a+b-c-d-h+g)\varepsilon^2 \leq \varepsilon\ .\end{aligned}$$

This covers all cases when $\boldsymbol{z} \in \{0, \varepsilon\}^5$ has three ε's among its coordinates.

Since $d \in [0,1]$, we have

$$\mathbb{f}\begin{pmatrix} \varepsilon \\ \varepsilon\,0\,\varepsilon \\ \varepsilon \end{pmatrix} = (1-d)\varepsilon \leq \varepsilon,$$

while $l_{30} \leq 1$ implies $2a + b - c + d - e - h + g \leq 0$, thus

$$\mathbb{f}\begin{pmatrix} & \varepsilon & \\ \varepsilon & \varepsilon & \varepsilon \\ & 0 & \end{pmatrix} = (a+b-c)\varepsilon^2 + (a+d-e)\varepsilon^2 + (a+g-h)\varepsilon^2 + (1-a)\varepsilon$$
$$= \varepsilon - (1-\varepsilon)\varepsilon a + (2a+b-c+d-e-h+g)\varepsilon^2 \leq \varepsilon \ ,$$

which concludes the proof of the first part of our claim.

For the second part, it suffices to use the conjugate rule. Indeed, if $\boldsymbol{z} \in [\varepsilon, 1]^5$, then $\mathbf{1} - \boldsymbol{z} \in [0, 1-\varepsilon]^5$ (see Eq. (2.6)), so, from the first part, we know that $\mathbb{f}^C(\mathbf{1} - \boldsymbol{z}) \leq 1 - \varepsilon$. But this also implies that $f(\boldsymbol{z}) = 1 - \mathbb{f}^C(\mathbf{1} - \boldsymbol{z}) \geq \varepsilon$. □

As a corollary of Theorem 4.3.2, we get that in the case of two-dimensional density-conserving ACCAs, the condition in Definition 4.3.1 is equivalent to:

$$(\exists T)\left(\forall \boldsymbol{x} \in \{0,1\}^C\right)\left(\begin{cases} \rho(\boldsymbol{x}) < \rho_0 \Rightarrow \mathbb{F}^T(\boldsymbol{x}) \in [0, \rho_0)^C \\ \rho(\boldsymbol{x}) > \rho_0 \Rightarrow \mathbb{F}^T(\boldsymbol{x}) \in (\rho_0, 1]^C \end{cases}\right) . \tag{4.3}$$

Indeed, if for some T all coordinates of $\boldsymbol{x}$ are smaller (resp. greater) than ρ_0, then for all $t \geq T$, all values $x^t_{i,j}$ are smaller (resp. greater) than ρ_0.

4.4 Conclusions

In this chapter, we defined the density classification problem in the context of ACCA. We see that the classical definition of the problem, that has no solution even for ACCAs, can be reformulated for it by a small modification of the output specification. The relaxation allows us to involve computer simulations in the next chapters and analyze the results.

Dynamics of One-Dimensional Outer-Totalistic ACCAs

5

In this chapter, we focus on one-dimensional ACCAs (with radius one) that are both outer-totalistic and legal. A combination of these two properties is very often used when modeling various phenomena. For example, it is a basic assumption in each "full of common sense" Life-like CA, *i.e.*, being similar to Conway's Game of Life [6, 73].

Among the 64 outer-totalistic ECAs only four are still outer-totalistic after the "fuzzification" process and show a very simple dynamics. Restricting our attention to the legal ones, we are left with only two ECAs: ECA0 and ECA204. The situation looks quite different for ACCAs. There are infinitely many legal outer-totalistic ACCAs and their dynamics are very diverse, which will be amply demonstrated in this chapter. To be more precise, we will uncover the dynamics of all such ACCAs.

5.1 Legal outer-totalistic ACCAs 41
5.2 Computer simulations . . 42
5.3 Dynamical properties of legal outer-totalistic one-dimensional ACCAs 45
5.4 The case $2d - c = 1$: the upper face 47
5.5 The case $2b+c < 1$ and $2d-c < 1$ 49
5.6 The case $2b+c \geq 1$ and $2d-c < 1$ 51
5.6.1 Some technical lemmata . 51
5.6.2 The case $(b, c, d) \in \mathcal{P}_0 \setminus (P_1P_3 \cup P_4P_6 \cup P_5P_7)$. . . 54
5.6.3 The case $(b, c, d) = (0.5, 1, d)$ for $0.5 \leq d < 1$. 57
5.6.4 The case $(b, c, d) = (0.5, 0, d)$ for $0 \leq d < 0.5$. 60
5.6.5 The case $(b, c, d) = (0, 1, d)$ for $0.5 \leq d < 1$. 62
5.7 Conclusions 67

5.1 Legal outer-totalistic ACCAs

The definition of an outer-totalistic CA and the theorem describing a necessary and sufficient condition for a one-dimensional ACCA to be outer-totalistic were already presented in Section 3.3.

Hence, we focus on a subclass of one-dimensional outer-totalistic ACCAs, known as legal ACCAs (Definition 2.6.1). According to Eqs. (3.5) and (3.6), the local rule of a legal outer-totalistic ACCA is given by

$$\mathbb{f}(x, y, z) = (d - c - b)(x + z)y + b(x + z) + cy \ , \tag{5.1}$$

where $0 \leq b \leq \frac{1}{2}$, $0 \leq c \leq 1$ and $c \leq 2d \leq c + 1$. These restrictions on the parameters b, c, d generate a parallelepiped $\mathcal{P}$ in the Cartesian coordinate system, which is shown in Figure 5.1.

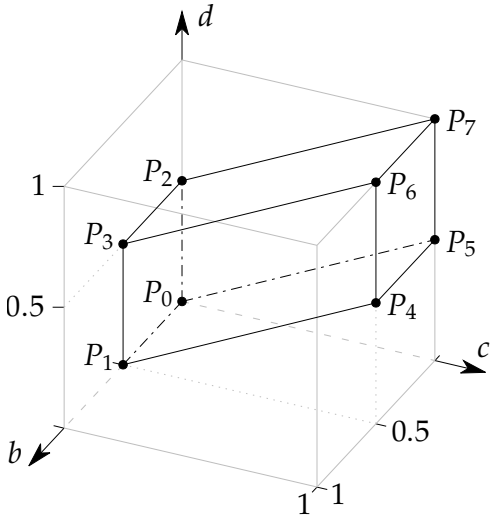


Figure 5.1: Parameterization of the set of legal outer-totalistic one-dimensional ACCAs.

As one can see, the parallelepiped $\mathcal{P}$ has eight vertices, thus there are exactly eight extreme points in the set of all legal outer-totalistic

one-dimensional ACCAs (see Table 5.1). The local rule of any legal outer-totalistic one-dimensional ACCA can be written as a convex combination of the local rules listed in the last column of Table 5.1. In the remainder of this chapter, we consider only legal outer-totalistic one-dimensional ACCAs, even if we do not emphasize it.

Table 5.1: Analytical expressions of the local rules corresponding to the extreme points in the set of legal outer-totalistic one-dimensional ACCAs.

Vertex	Corresponding local rule
$P_0 = (0,0,0)$	$\mathbb{f}(x,y,z) = 0$ (ECA 0)
$P_1 = (\frac{1}{2},0,0)$	$\mathbb{f}(x,y,z) = \frac{1}{2}(x+z)(1-y)$
$P_2 = (0,0,\frac{1}{2})$	$\mathbb{f}(x,y,z) = \frac{1}{2}(x+z)y$
$P_3 = (\frac{1}{2},0,\frac{1}{2})$	$\mathbb{f}(x,y,z) = \frac{1}{2}(x+z)$
$P_4 = (\frac{1}{2},1,\frac{1}{2})$	$\mathbb{f}(x,y,z) = y + (x+z)(\frac{1}{2}-y)$
$P_5 = (0,1,\frac{1}{2})$	$\mathbb{f}(x,y,z) = y - \frac{1}{2}(x+z)y$
$P_6 = (\frac{1}{2},1,1)$	$\mathbb{f}(x,y,z) = y + \frac{1}{2}(x+z)(1-y)$
$P_7 = (0,1,1)$	$\mathbb{f}(x,y,z) = y$ (ECA 204)

5.2 Computer simulations

Our goal is to examine the dynamics of all legal outer-totalistic ACCAs, more specifically, we want to determine how these dynamics evolve when starting from binary configurations. We concentrate on such initial configurations for two reasons. Firstly, many problems concerning cellular automata are centered on binary configurations (as, for example, the density classification problem). Secondly, we want to deal with a finite set of initial configurations to be able to examine all of them using computer simulations.

Our investigation started with very detailed simulations, described below, whose results facilitated to formulate preliminary hypotheses (depending on the range of the parameters b, c and d), some of which could be easily confirmed theoretically, while other ones required often additional simulations, revealing the underlying mechanism of a given ACCA in detail. This was essential in order to be able to choose the proper analytical tools and mathematical proof methods. As a result, it became possible to describe and theoretically confirm the dynamics of all legal outer-totalistic one-dimensional ACCAs. Below, we present the experimental set-up for the first part of the simulations.

If the number of cells N is given, then X_N denotes the set of all binary configurations of length N, *i.e.*, $X_N = \{0,1\}^N$. Additionally, we use the notations X_N^0, X_N^1 and $X_N^{0,1}$ to refer to the set X_N without 0^N, 1^N or both of these elements, respectively, *i.e.*,

$$\begin{aligned} X_N^0 &= X_N \setminus \{0^N\} , \\ X_N^1 &= X_N \setminus \{1^N\} , \\ X_N^{0,1} &= X_N \setminus \{0^N, 1^N\} . \end{aligned}$$

In the first part of our experiments, we sampled the set of all legal outer-totalistic one-dimensional ACCAs parameterized by (b, c, d) by varying the parameter c from 0 to 1 with a step size of 0.1 and – taking into account Theorem 3.3.1 – the parameters b, d from 0 to 0.5 and from $0.5c$ to $0.5c + 0.5$, respectively, with a step size of 0.05. For each such set of parameters, we defined the local rule $\mathbb{f}$ by the expression in Eq. (5.1), resulting in a set $\mathcal{U}$ containing 1331 rules.

Because we did not want to be dependent on randomly selected configurations, we chose small N and examined all binary configurations of this length. For every local rule $\mathbb{f} \in \mathcal{U}$, we carried out the following simulations. For each $\boldsymbol{x} \in X_N$, we ran the corresponding global rule $\mathbb{F}$ for $\Lambda = 10000$ iterations and classified $\boldsymbol{x}$, by checking the output configurations $\mathbb{F}^{\Lambda}(\boldsymbol{x})$, $\mathbb{F}^{\Lambda-1}(\boldsymbol{x})$, $\mathbb{F}^{\Lambda-2}(\boldsymbol{x})$, to the first group $G_i^{(N)}$ with a matching condition:

$G_0^{(N)}$**:** all cell states are close to 0 and configurations $\mathbb{F}^{\Lambda}(\boldsymbol{x})$ and $\mathbb{F}^{\Lambda-1}(\boldsymbol{x})$ are almost the same, *i.e.*,

$$\max(\mathbb{F}^{\Lambda}(\boldsymbol{x})) < \epsilon \text{ and } \|\mathbb{F}^{\Lambda}(\boldsymbol{x}) - \mathbb{F}^{\Lambda-1}(\boldsymbol{x})\| < \eta \ .$$

In this case, we conjecture that $\mathbb{F}^t(\boldsymbol{x})$ tends to 0^N.

$G_1^{(N)}$**:** all cell states are close to 1 and configurations $\mathbb{F}^{\Lambda}(\boldsymbol{x})$ and $\mathbb{F}^{\Lambda-1}(\boldsymbol{x})$ are almost the same, *i.e.*,

$$\min(\mathbb{F}^{\Lambda}(\boldsymbol{x})) > 1 - \epsilon \text{ and } \|\mathbb{F}^{\Lambda}(\boldsymbol{x}) - \mathbb{F}^{\Lambda-1}(\boldsymbol{x})\| < \eta \ .$$

In this case, we conjecture that $\mathbb{F}^t(\boldsymbol{x})$ tends to 1^N.

$G_2^{(N)}$**:** all cell states are close to the same value and configurations $\mathbb{F}^{\Lambda}(\boldsymbol{x})$ and $\mathbb{F}^{\Lambda-1}(\boldsymbol{x})$ are almost the same, *i.e.*,

$$\max(\mathbb{F}^{\Lambda}(\boldsymbol{x})) - \min(\mathbb{F}^{\Lambda}(\boldsymbol{x})) < \epsilon \text{ and } \|\mathbb{F}^{\Lambda}(\boldsymbol{x}) - \mathbb{F}^{\Lambda-1}(\boldsymbol{x})\| < \eta \ .$$

In this case, we conjecture that $\mathbb{F}^t(\boldsymbol{x})$ tends to c^N, for some $c \in (0, 1)$ (the constant c may depend on $\boldsymbol{x}$).

$G_3^{(N)}$**:** all cell states are close to the same value and configurations $\mathbb{F}^{\Lambda}(\boldsymbol{x})$ and $\mathbb{F}^{\Lambda-2}(\boldsymbol{x})$ are almost the same, *i.e.*,

$$\max(\mathbb{F}^{\Lambda}(\boldsymbol{x})) - \min(\mathbb{F}^{\Lambda}(\boldsymbol{x})) < \epsilon \text{ and } \|\mathbb{F}^{\Lambda}(\boldsymbol{x}) - \mathbb{F}^{\Lambda-2}(\boldsymbol{x})\| < \eta \ .$$

In this case, we conjecture that $\mathbb{F}^{2t}(\boldsymbol{x})$ tends to c_1^N and $\mathbb{F}^{2t+1}(\boldsymbol{x})$ tends to c_2^N, for some $c_1, c_2 \in [0, 1]$ (a convergence with period 2).

$G_4^{(N)}$**:** configurations $\mathbb{F}^{\Lambda}(\boldsymbol{x})$ and $\mathbb{F}^{\Lambda-1}(\boldsymbol{x})$ are almost the same, *i.e.*,

$$\|\mathbb{F}^{\Lambda}(\boldsymbol{x}) - \mathbb{F}^{\Lambda-1}(\boldsymbol{x})\| < \eta \ .$$

In this case, we assume that $\mathbb{F}^t(\boldsymbol{x})$ tends to some fixed configuration.

$G_5^{(N)}$**:** all other configurations.

We will drop the upper index (N) unless confusion is possible.

For the reported experiments, we chose $\epsilon = 0.001$ and $\eta = 0.01$. As a result, for every local rule $\mathbb{f} \in \mathcal{U}$, we obtained 12 subsets of binary configurations:

$$G_0^{(10)}, \ldots, G_5^{(10)} \text{ and } G_0^{(11)}, \ldots, G_5^{(11)} \ ,$$

satisfying

$$G_0^{(10)} \cup G_1^{(10)} \cup \ldots \cup G_5^{(10)} = X_{10} \text{ and } G_0^{(11)} \cup G_1^{(11)} \cup \ldots \cup G_5^{(11)} = X_{11} \, .$$

On this basis, we formulated the preliminary hypothesis on the dynamics of $\mathbb{f}$.

For example, for the local rule

$$\mathbb{f}(x, y, z) = 0.1(x + z)y + 0.5y \, ,$$

parameterized by the point

$$(b, c, d) = (0, 0.5, 0.6) \, ,$$

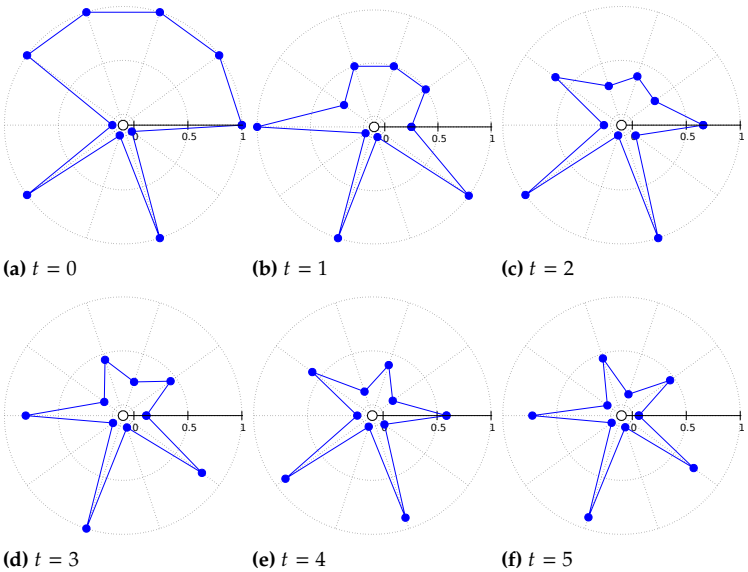
the simulations yield $G_0^{(10)} = X_{10}$ and $G_0^{(11)} = X_{11}$ (while the other subsets are empty). Thus, it is obvious that we conjecture that for each binary configuration $\boldsymbol{x} \in X_N$, it holds that $\mathbb{F}^t(\boldsymbol{x})$ tends to 0^N.

As mentioned above, the results of these first experiments allowed to detect the nature of many legal outer-totalistic ACCAs. Moreover, it is not hard to confirm these observations theoretically (see, for example, Subsection 5.5). However, for some local rules we were forced to perform additional simulations. In particular, we varied N, ϵ, η, and used various data exploration techniques that allow for tracking both the positions and values of $\min(\mathbb{F}^t(\boldsymbol{x}))$ and $\max(\mathbb{F}^t(\boldsymbol{x}))$. In some cases, we relied on a visualization of the dynamics.

To visualize the dynamics of an ACCA, we use the radial view of CCAs presented in Section 2.9. The radial representations at subsequent time steps allow to observe the evolution of a given CCA and gain some intuition about the rules that govern it. For example, Figure 5.2 shows the dynamics of the ACCA with the local rule

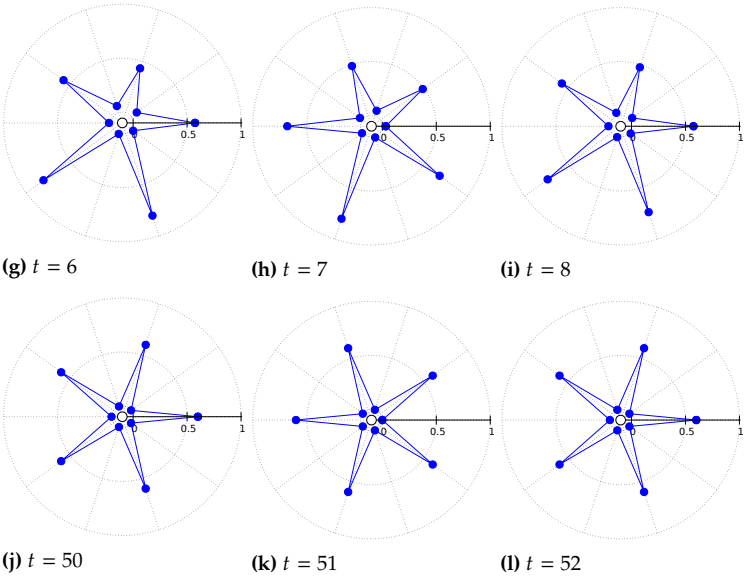
$$\mathbb{f}(\boldsymbol{x}) = -0.25(x + z)y + 0.5(x + z) + 0.25y \, ,$$

parameterized by the point $(b, c, d) = (0.5, 0, 0.25)$, starting from the



(a) $t = 0$ **(b)** $t = 1$ **(c)** $t = 2$

(d) $t = 3$ **(e)** $t = 4$ **(f)** $t = 5$



(g) $t = 6$ **(h)** $t = 7$ **(i)** $t = 8$

(j) $t = 50$ **(k)** $t = 51$ **(l)** $t = 52$

Figure 5.2: Visualization of the dynamics of an ACCA with the local rule parameterized by the point $(b, c, d) = (0.5, 0, 0.25)$ starting from the initial configuration $\boldsymbol{x} = (1)^4(10)^3$. The subsequent radial views show $\mathbb{F}^t(\boldsymbol{x})$ for given t.

initial configuration $\boldsymbol{x} = 1^4(10)^3$. The simulation suggests a convergence with period 2: $\mathbb{F}^{2t}(\boldsymbol{x}) \to (\alpha 0)^5$ and $\mathbb{F}^{2t+1}(\boldsymbol{x}) \to (0\alpha)^5$, for some $\alpha \in (0, 1)$. As we will see further on, Theorem 5.6.7 will confirm this observation.

5.3 Dynamical properties of legal outer-totalistic one-dimensional ACCAs

In the next sections, we provide a detailed description of the dynamics of all legal outer-totalistic ACCAs. We start with the following simple, yet crucial observation concerning local rules parameterized by points of $\mathcal{P}$ not lying on the left face $(P_0P_1P_3P_2)$, back face $(P_0P_5P_7P_2)$ and upper face $(P_2P_3P_6P_7)$, *i.e.*, points $(b, c, d) \in \mathcal{P}$ satisfying $b, c > 0$ and $2d - c < 1$.

Lemma 5.3.1 *Let $\mathbb{f}$ be the local rule parameterized by a point $(b, c, d) \in \mathcal{P}$ and let $\mathbb{F}$ be the corresponding global rule. Suppose that $b, c > 0$ and $2d - c < 1$.*

(p1) *Let $(x, y, z) \in [0, 1]^3$. Then $\mathbb{f}(x, y, z) = 0$ only for $(x, y, z) = (0, 0, 0)$ and, additionally, for $(x, y, z) = (1, 1, 1)$ if $c = 2d$. Furthermore, $\mathbb{f}(x, y, z) = 1$ only for $(x, y, z) = (0, 1, 0)$ if $c = 1$ and $(x, y, z) = (1, 0, 1)$ if $b = \frac{1}{2}$.*

(p2) *For any $\boldsymbol{x} \in X_N^{0,1}$ and any $t \geq 2$, the configuration $\mathbb{F}^t(\boldsymbol{x})$ does not contain any* 1*s.*

(p3) *For any $\boldsymbol{x} \in X_N^{0,1}$ (if $(b, c, d) \neq P_4$ or N is odd) and for any $\boldsymbol{x} \in X_N^{0,1} \setminus \left\{(01)^{\frac{N}{2}}, (10)^{\frac{N}{2}}\right\}$ (if $(b, c, d) = P_4$ and N is even), if $t > \frac{N}{2} + 2$, then the configuration $\mathbb{F}^t(\boldsymbol{x})$ does not contain any* 0*s.*

Proof.

(p1) Let $0 < b \leq \frac{1}{2}$, $0 < c \leq 1$ and $c \leq 2d < c + 1$. Note that according

to Eq. (5.1), it holds that

$$\begin{aligned} \mathfrak{f}(0,0,0) &= 0 \ , \\ \mathfrak{f}(0,1,0) &= c \ , \\ \mathfrak{f}(0,0,1) &= \mathfrak{f}(1,0,0) = b \ , \\ \mathfrak{f}(1,0,1) &= 2b \ , \\ \mathfrak{f}(0,1,1) &= \mathfrak{f}(1,1,0) = d \ , \\ \mathfrak{f}(1,1,1) &= 2d - c \ . \end{aligned} \tag{5.2}$$

Among the values different from $\mathfrak{f}(0,0,0)$, only $\mathfrak{f}(1,1,1)$ may be zero if $2d - c = 0$. Moreover, the vertices $(0,0,0)$ and $(1,1,1)$ do not lie on a common edge in $[0,1]^3$, thus it is not possible to get $\mathfrak{f}(x,y,z) = 0$ if (x,y,z) is neither $(0,0,0)$ nor $(1,1,1)$. This follows directly from the fact that $\mathfrak{f}$ is affine w.r.t. each variable and the minimal value of $\mathfrak{f}$ on $[0,1]^3$ is zero. Similarly, $\mathfrak{f}(x,y,z)$ only takes value 1 for $(0,1,0)$ if $c = 1$, or $(1,0,1)$ if $b = \frac{1}{2}$.

(p2) Suppose that $\boldsymbol{x} \in X_N^{0,1}$, $t \geq 2$, but $\mathbb{F}^t(\boldsymbol{x})$ contains 1. If so, then according to (p1), $\mathbb{F}^{t-1}(\boldsymbol{x})$ has to contain a string 01 and, consequently, $\mathbb{F}^{t-2}(\boldsymbol{x})$ contains a string $xyzu$ such that $\mathfrak{f}(x,y,z) = 0$ and $\mathfrak{f}(y,z,u) = 1$. However, the first equality implies $y = z$, while the second one is possible only when $y \neq z$. This contradiction shows that $\mathbb{F}^t(\boldsymbol{x})$ cannot contain any 1s if $t - 2 \geq 0$.

(p3) Suppose that $\boldsymbol{x} \in X_N^{0,1}$. From (p2) we know that $\mathbb{F}^2(\boldsymbol{x})$ does not contain any 1s. According to (p1), if $\mathbb{F}^2(\boldsymbol{x}) = 0^N$, then $\mathbb{F}^1(\boldsymbol{x}) = 0^N$, or $\mathbb{F}^1(\boldsymbol{x}) = 1^N$ and $c = 2d$. As the first case is impossible (since $\boldsymbol{x} \neq 0^N$ and $\boldsymbol{x} \neq 1^N$), it has to hold that $\mathbb{F}^1(\boldsymbol{x}) = 1^N$ and $c = 2d$. However, $\mathbb{F}^1(\boldsymbol{x}) = 1^N$ only when N is even, $\boldsymbol{x} = (01)^{\frac{N}{2}}$ or $\boldsymbol{x} = (10)^{\frac{N}{2}}$, and both $b = \frac{1}{2}$ and $c = 1$. Thus, if $(b,c,d) \neq P_4$ or N is odd, then $\mathbb{F}^2(\boldsymbol{x}) \neq 0^N$. The same holds when $(b,c,d) = P_4$, N is even but $\boldsymbol{x} \neq (01)^{\frac{N}{2}}$ and $\boldsymbol{x} \neq (10)^{\frac{N}{2}}$. If so, then in each subsequent time step, the length of every string of 0s in $\boldsymbol{x}$ is reduced by one on each side, hence $\mathbb{F}^{2+s}(\boldsymbol{x})$ has no 0 for $s > \frac{N}{2}$.

□

Since we consider legal ACCAs, it holds that $\mathbb{F}^t(0^N) = 0^N$ for any $t > 0$. Moreover, as we will see, for many $(b,c,d) \in \mathcal{P}$ and many $\boldsymbol{x} \in X_N$, it holds that $\mathbb{F}^t(\boldsymbol{x}) \to 0^N$. The following lemma characterizes the homogeneous configurations $s^N \in X_N$ that may be the limit of some sequence $\left(\mathbb{F}^t(\boldsymbol{x})\right)_{t\in\mathbb{N}}$ when $b + c - d \neq 0$.

Lemma 5.3.2 *Let $\mathfrak{f}$ be the local rule parameterized by a point $(b,c,d) \in \mathcal{P}$ and let $\mathbb{F}$ be the corresponding global rule. Suppose that $b + c - d \neq 0$. If for some $\boldsymbol{x} \in X_N$ it holds that $\lim_{t\to\infty} \mathbb{F}^t(\boldsymbol{x}) = s^N$, then $s = 0$ or $s = \frac{2b+c-1}{2(b+c-d)}$.*

Proof. If $\lim_{t\to\infty} \mathbb{F}^t(\boldsymbol{x}) = s^N$, then $\mathfrak{f}(s,s,s) = s$. Hence, according to Eq. (5.1) we obtain

$$2(d - c - b)s^2 + (2b + c)s = s \ . \tag{5.3}$$

Since $b + c - d \neq 0$, Eq. (5.3) has only two solutions: $s = 0$ or $s = \frac{2b+c-1}{2(b+c-d)}$, which proves our claim. □

Note that although $\frac{2b+c-1}{2(b+c-d)}$ is bounded from above by 1, it still might be negative, in which case the only possible homogeneous limit of $\mathbb{F}^t(\boldsymbol{x})$ is 0^N.

5.4 The case $2d - c = 1$: the upper face

We start our investigation with local rules that satisfy both

$$l_0 = \mathbb{f}(0,0,0) = 0 \ ,$$
$$l_7 = \mathbb{f}(1,1,1) = 2d - c = 1 \ ,$$

i.e., double-legal local rules. These rules are parameterized by points lying on the upper face $P_2P_3P_6P_7$ of the parallelepiped $\mathcal{P}$ (see Figure 5.3a). In this case, the expression of the local rule $\mathbb{f}$ reduces to

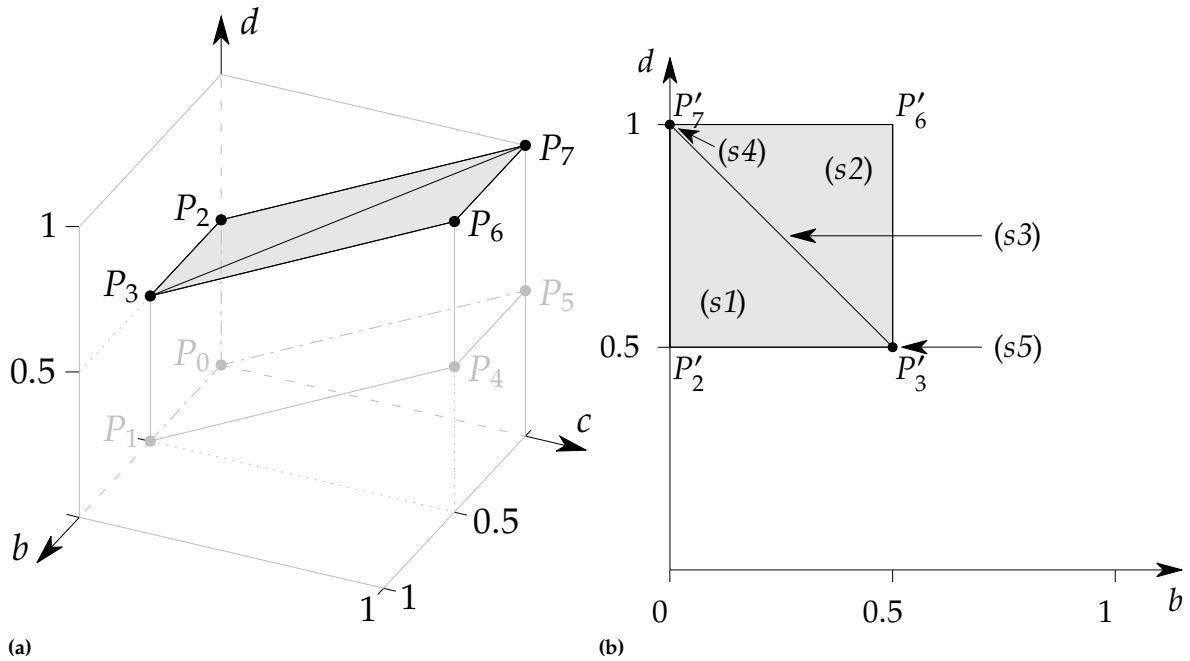


Figure 5.3: Parameterization of the set of double-legal outer-totalistic ACCAs: in the parallelepiped $\mathcal{P}$ (a) in the square $\mathcal{S}$ (b).

$$\mathbb{f}(x,y,z) = (1-b-d)(x+z)y + b(x+z) + (2d-1)y \ ,$$

where $0 \leq b \leq \frac{1}{2}$ and $\frac{1}{2} \leq d \leq 1$. Thus, the set of double-legal outer-totalistic ACCAs can also be parameterized by the points of the square $P'_2P'_3P'_6P'_7$ shown in Figure 5.3b and denoted in the remainder by $\mathcal{S}$. Note that if $\mathbb{f}$ is double-legal, then also $\mathbb{f}^C$ is double-legal. Moreover, the points that parameterize $\mathbb{f}$ and $\mathbb{f}^C$ are symmetric w.r.t. the diagonal $P'_3P'_7$. As the behavior of $\mathbb{F}^C$ is strongly related to the behavior of $\mathbb{F}$, it is sufficient to consider the case $b + d \leq 1$.

In Table 5.2, we list a few points (b, c, d) corresponding to double-legal outer-totalistic ACCAs with distinct dynamics and indicate the corresponding groups of initial configurations identified through simulation. These distinct dynamics will be discussed analytically further on.

Table 5.2: Examples of points (b, c, d) representing five distinct dynamics observed in the simulation of double-legal outer-totalistic ACCAs lying on the upper face $P'_2P'_3P'_6P'_7$ of the parallelepiped $\mathcal{P}$.

(b, c, d)	$N = 10$	$N = 11$
$(0.2, 0.4, 0.7)$	$G_0 = X^1_{10}$ $G_1 = \{1^{10}\}$	$G_0 = X^1_{11}$ $G_1 = \{1^{11}\}$
$(0.2, 0.4, 0.9)$	$G_0 = \{0^{10}\}$ $G_1 = X^0_{10}$	$G_0 = \{0^{11}\}$ $G_1 = X^0_{11}$
$(0.2, 0.6, 0.8)$	$G_0 = \{0^{10}\}$ $G_1 = \{1^{10}\}$ $G_2 = X^{0,1}_{10}$	$G_0 = \{0^{11}\}$ $G_1 = \{1^{11}\}$ $G_2 = X^{0,1}_{11}$
$(0.0, 1.0, 1.0)$	$G_0 = \{0^{10}\}$ $G_1 = \{1^{10}\}$ $G_4 = X^{0,1}_{10}$	$G_0 = \{0^{11}\}$ $G_1 = \{1^{11}\}$ $G_4 = X^{0,1}_{11}$
$(0.5, 0.0, 0.5)$	$G_0 = \{0^{10}\}$ $G_1 = \{1^{10}\}$ $G_2 \neq \emptyset$ $G_5 \neq \emptyset$	$G_0 = \{0^{11}\}$ $G_1 = \{1^{11}\}$ $G_2 = X^{0,1}_{11}$

Theorem 5.4.1 *Let $\mathbb{f}$ be the local rule parameterized by a point $(b, d) \in \mathcal{S}$ and let $\mathbb{F}$ be the corresponding global rule.*

(s1) *If $b + d < 1$, then for any initial configuration $\boldsymbol{x} \in X^1_N$, it holds that*

$$\lim_{t\to\infty} \mathbb{F}^t(\boldsymbol{x}) = 0^N \ ,$$

while $\mathbb{F}^t(1^N) = 1^N$ for each $t > 0$.

(s2) *If $b + d > 1$, then for any initial configuration $\boldsymbol{x} \in X^0_N$, it holds that*

$$\lim_{t\to\infty} \mathbb{F}^t(\boldsymbol{x}) = 1^N \ ,$$

while $\mathbb{F}^t(0^N) = 0^N$ for each $t > 0$.

(s3) *If $b + d = 1$ and $0 < b < \frac{1}{2}$, then for any initial configuration $\boldsymbol{x} \in X_N$, it holds that*

$$\lim_{t\to\infty} \mathbb{F}^t(\boldsymbol{x}) = (\rho(\boldsymbol{x}))^N \ .$$

(s4) *If $b = 0$ and $d = 1$ (i.e., $(b, d) = P'_7$), then $\mathbb{f}$ is the identity rule, so each configuration $\boldsymbol{x} \in X_N$ is a fixed point of $\mathbb{F}$.*

(s5) *If $b = d = \frac{1}{2}$ (i.e., $(b, d) = P'_3$), then for any initial configuration $\boldsymbol{x} \in X_N$, it holds that*

- *if N is odd, then*

$$\lim_{t\to\infty} \mathbb{F}^t(\boldsymbol{x}) = (\rho(\boldsymbol{x}))^N \ ;$$

► *if* N *is even, then*

$$\begin{cases} \lim_{t\to\infty} \mathbb{F}^{2t}(\boldsymbol{x}) = (\rho_0(\boldsymbol{x})\rho_1(\boldsymbol{x}))^{\frac{N}{2}} \\ \lim_{t\to\infty} \mathbb{F}^{2t+1}(\boldsymbol{x}) = (\rho_1(\boldsymbol{x})\rho_0(\boldsymbol{x}))^{\frac{N}{2}} \end{cases} ,$$

where

$$\rho_0(\boldsymbol{x}) = \frac{2}{N}(x_0 + x_2 + \ldots + x_{N-2}) ,$$
$$\rho_1(\boldsymbol{x}) = \frac{2}{N}(x_1 + x_3 + \ldots + x_{N-1}) .$$

Proof. Consider $(b, d) \in \mathcal{S}$ and let $\boldsymbol{x} \in X_N^1$. First, let us note that if $b + d < 1$, then $x, y, z \leq M \leq 1$ implies $\mathbb{f}(x, y, z) \leq M$. Indeed,

$$\begin{aligned} \mathbb{f}(x, y, z) &= (1 - b - d)(x + z)y + b(x + z) + (2d - 1)y \\ &\leq (1 - b - d)(x + z) + b(x + z) + (2d - 1)y \\ &\leq (1 - b - d)2M + 2bM + (2d - 1)M = M . \end{aligned}$$

Moreover, from the above calculations, it is obvious that if at least one variable x, y, z is strictly smaller than 1, then also $\mathbb{f}(x, y, z) < 1$. This means that if for some $t_0 \in \mathbb{N}$ the configuration $\mathbb{F}^{t_0}(\boldsymbol{x})$ does not contain any 1s, *i.e.*, $M = \max(\mathbb{F}^{t_0}(\boldsymbol{x})) < 1$, then for each $t \geq t_0$ and $0 \leq n \leq N - 1$, it holds that $x_n^t \leq M$. Hence

$$\begin{aligned} \sigma(\mathbb{F}^{t+1}(\boldsymbol{x})) &= \sum_{n=0}^{N-1} x_n^{t+1} \\ &= \sum_{n=0}^{N-1} \left((1 - b - d)(x_{n-1}^t + x_{n+1}^t)x_n^t + b(x_{n-1}^t + x_{n+1}^t) + (2d - 1)x_n^t\right) \\ &\leq \left(\sum_{n=0}^{N-1}(1 - b - d)2Mx_n^t\right) + (2b + 2d - 1)\sigma(\mathbb{F}^t(\boldsymbol{x})) \\ &= (1 - b - d)2M\sigma(\mathbb{F}^t(\boldsymbol{x})) + (2b + 2d - 1)\sigma(\mathbb{F}^t(\boldsymbol{x})) \\ &= (1 - 2(1 - b - d)(1 - M))\,\sigma(\mathbb{F}^t(\boldsymbol{x})) . \end{aligned}$$

This means that $\sigma(\mathbb{F}^t(\boldsymbol{x}))$ tends to zero (since $1 - 2(1 - b - d)(1 - M) < 1$), which concludes the proof of statement (s1). As mentioned before, statement (s2) is a direct consequence of (s1), since local rules parameterized by points $(b, d) \in \mathcal{S}$ with $b + d > 1$ are conjugate to those with $b + d < 1$.

If $b + d = 1$, then $\mathbb{f}(x, y, z) = b(x + z) + (1 - 2b)y$ and $\mathbb{f}$ is density-conserving, *i.e.*, for any $\boldsymbol{x} \in X_N$, it holds that

$$\rho(\boldsymbol{x}) = \rho(\mathbb{F}(\boldsymbol{x})) = \rho(\mathbb{F}^2(\boldsymbol{x})) = \rho(\mathbb{F}^3(\boldsymbol{x})) = \ldots .$$

We will present in detail the dynamics of such ACCAs in Chapter 6, including the proof of statements (s3) and (s5). Statement (s4) is obvious. □

5.5 The case $2b + c < 1$ and $2d - c < 1$

The points of $\mathcal{P}$ satisfying $2b + c < 1$ and $2d - c < 1$ are shown in Figure 5.4.

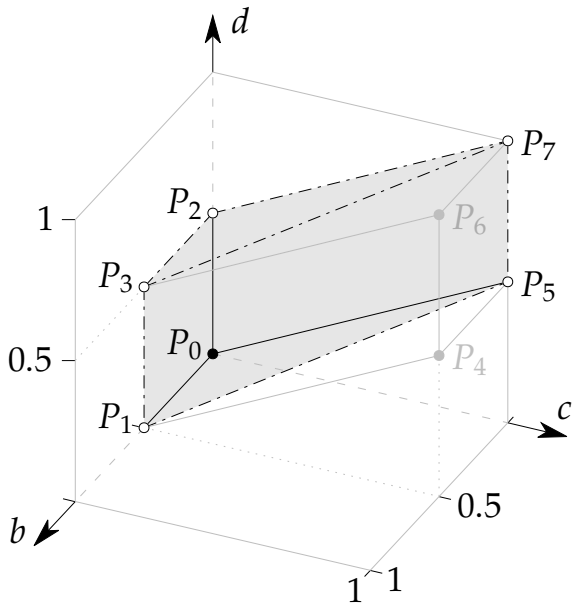


Figure 5.4: Parameterization of the set of legal outer-totalistic one-dimensional ACCAs in the case $2b + c < 1$ and $2d - c < 1$.

The dynamics of ACCAs whose local rules are parameterized by these points is very simple. The state of each cell tends to zero, irrespective of the initial configuration. The following theorem formally characterizes the behavior of such CAs.

Theorem 5.5.1 *Let $\mathbb{f}$ be the local rule parameterized by a point $(b, c, d) \in \mathcal{P}$ and let $\mathbb{F}$ be the corresponding global rule. If $2b + c < 1$ and $2d - c < 1$, then for any initial configuration $\boldsymbol{x} \in X_N$, it holds that $\lim_{x\to\infty} \mathbb{F}^t(\boldsymbol{x}) = 0^N$.*

Proof. Let us choose $N \in \mathbb{N}$ and $\boldsymbol{x} \in X_N$. For any $t \geq 0$, it holds that

$$\begin{aligned}
\sigma\left(\mathbb{F}^{t+1}(\boldsymbol{x})\right) &= \sum_{n=0}^{N-1} x_n^{t+1} \\
&= \sum_{n=0}^{N-1} \left[(d-c-b)(x_{n-1}^t + x_{n+1}^t)x_n^t + b(x_{n-1}^t + x_{n+1}^t) + cx_n^t\right] \\
&= \left(\sum_{n=0}^{N-1}(d-c-b)(x_{n-1}^t + x_{n+1}^t)x_n^t\right) + (2b+c)\sigma\left(\mathbb{F}^t(\boldsymbol{x})\right) \ .
\end{aligned}$$

Hence, if $d - c - b < 0$, then

$$\sigma(\mathbb{F}^{t+1}(\boldsymbol{x})) \leq (2b + c)\sigma(\mathbb{F}^t(\boldsymbol{x})) \ .$$

If $d - c - b \geq 0$, then

$$\begin{aligned}
\sum_{n=0}^{N-1}(d-c-b)(x_{n-1}^t + x_{n+1}^t)x_n^t &\leq \sum_{n=0}^{N-1}(d-c-b)(x_{n-1}^t + x_{n+1}^t) \\
&= 2(d-c-b)\sigma(\mathbb{F}^t(\boldsymbol{x})) \ ,
\end{aligned}$$

thus, in this case,

$$\begin{aligned}
\sigma(\mathbb{F}^{t+1}(\boldsymbol{x})) &\leq 2(d-c-b)\sigma(\mathbb{F}^t(\boldsymbol{x})) + (2b+c)\sigma(\mathbb{F}^t(\boldsymbol{x})) \\
&= (2d-c)\sigma(\mathbb{F}^t(\boldsymbol{x})) \ .
\end{aligned}$$

Since $2b + c < 1$ and $2d - c < 1$, we see that, in both cases, $\sigma(\mathbb{F}^t(\boldsymbol{x}))$ tends to zero, which concludes the proof. □

5.6 The case $2b + c \geq 1$ and $2d - c < 1$

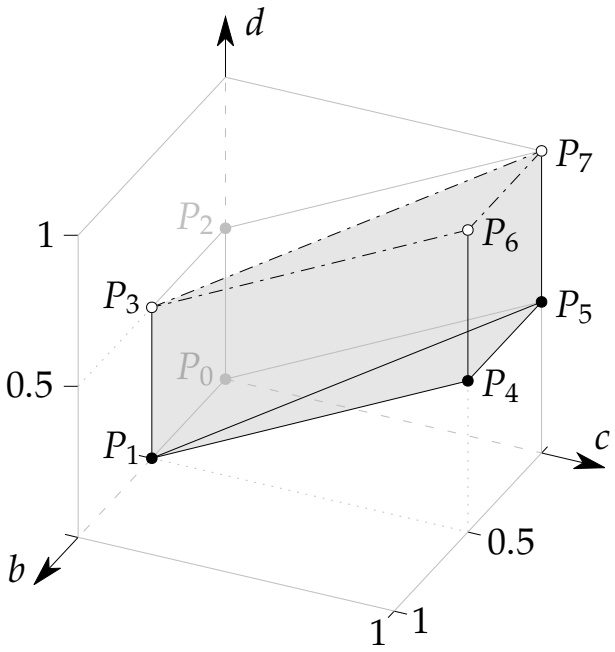


Figure 5.5: The polyhedron $\mathcal{P}_0$.

To conclude our investigation, we explore the dynamics of the local rules parameterized by points belonging to the polyhedron $P_1P_4P_5P_3P_6P_7$ without the upper face $P_3P_6P_7$, denoted by $\mathcal{P}_0$ and shown in Figure 5.5. This set is explicitly given as

$$\mathcal{P}_0 = \{(b, c, d) \in \mathcal{P} \mid 2b + c \geq 1 \text{ and } 2d - c < 1\} .$$

In this subsection, the expression $\frac{2b+c-1}{2(b+c-d)}$ will be denoted by λ and will turn out to be a key parameter. Note that if $(b, c, d) \in \mathcal{P}_0$, then it holds in particular that $b + c - d > 0$, hence λ is well defined and lies in $[0, 1]$.

This section is organized as follows. We start with some technical lemmata. Then we describe the dynamics of the rules parameterized by points belonging to $\mathcal{P}_0$, except for the three edges P_1P_3, P_4P_6 and P_5P_7. Next, we consider the edges P_4P_6, P_1P_3 and P_5P_7 one by one. The latter case is the most complicated one, since the rules parameterized by points from P_5P_7 exhibit the most surprising behavior.

5.6.1 Some technical lemmata

In order to prove the main theorems in this section, we introduce a few lemmata.

Lemma 5.6.1 *Let* $\mathbb{f}$ *be the local rule parameterized by a point* $(b, c, d) \in \mathcal{P}_0$.

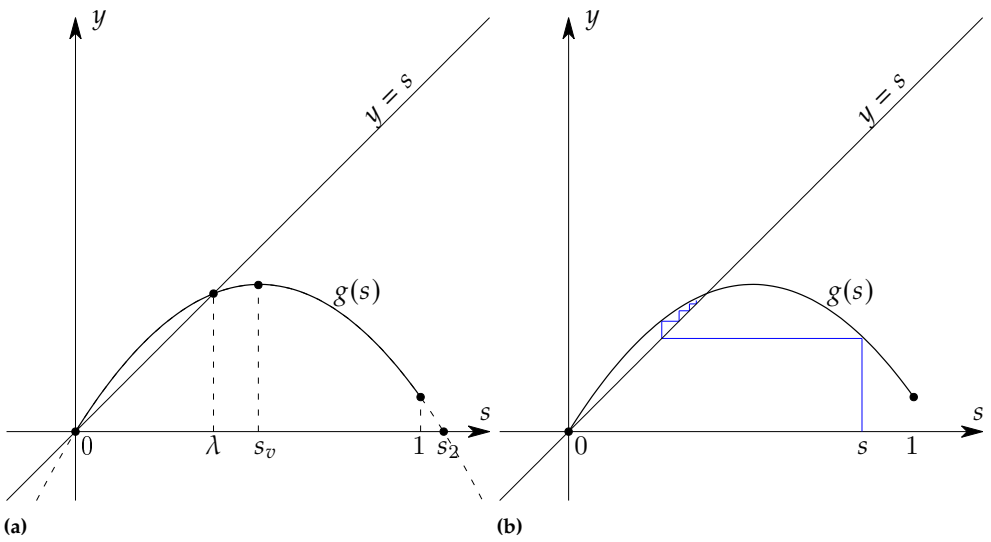


Figure 5.6: The graph of the function g in the case $2b + c > 1$. Location of the characteristic points (a) and attractor for an example point $s \in (0, 1)$ (b).

The function $g : [0, 1] \to \mathbb{R}$ *defined by*

$$g(s) := \mathbb{f}(s, s, s) = -2(b + c - d)s^2 + (2b + c)s$$

has the following two properties;

(g1) *Let* $s_\star \in [\lambda, 1]$. *If* $0 \le s \le s_\star$, *then* $0 \le g(s) \le s_\star$.
(g2) *If* $0 < s < 1$, *then* $\lim_{t\to\infty} g^t(s) = \lambda$, *where*

$$g^t(s) = \underbrace{g \circ g \circ \ldots \circ g}_{t \text{ times}}(s) .$$

Proof. Assume first that $2b + c > 1$, then the graph of the function g is shown in Figure 5.6a (under the proviso that s_v may lie to the right of 1 or s_2 may be equal to 1). Indeed, the function g is a quadratic function with roots $s_1 = 0$ and $s_2 = \frac{2b+c}{2(b+c-d)}$, restricted to the domain $[0, 1]$. It has two fixed points, namely 0 and λ. Since $c \le 2d \le c + 1$, it holds that $\lambda \le 1 \le s_2$. Directly from the graph of g, we can conclude that (g1) holds, since for $s \ge \lambda$ we have $g(s) \le s$. Moreover, the fixed point $s_1 = 0$ is a repeller, while λ is an attractor with basin including $(0, 1)$. This implies (g2).

Next, assume that $2b + c = 1$. In that case, it holds that $\lambda = 0$ and the line $y = s$ is tangent to the graph of g at $s_1 = 0$ (see Figure 5.7) and one can see that 0 is an attractor with basin $[0, 1]$. □

As a simple consequence of Lemma 5.6.1, we get the following description of the orbits for the initial configuration 1^N.

Lemma 5.6.2 *Let* $\mathbb{f}$ *be the local rule parameterized by a point* $(b, c, d) \in \mathcal{P}_0$ *and let* $\mathbb{F}$ *be the corresponding global rule. If* $c = 2d$ *(i.e.,* (b, c, d) *belongs to the lower face* $P_1P_4P_5$*), then* $\mathbb{F}^t(1^N) = 0^N$, *for each* $t \ge 1$; *otherwise,* $\mathbb{F}^t(1^N) \xrightarrow{t\to\infty} \lambda^N$.

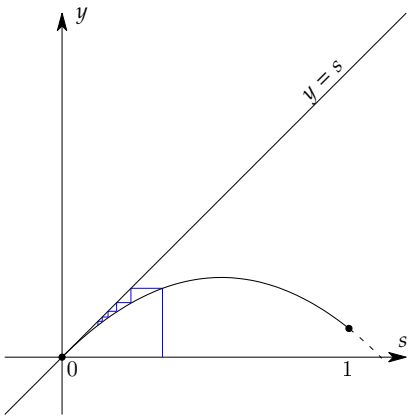


Figure 5.7: The graph of the function g in the case $2b + c = 1$. The line $y = s$ is tangent to the graph of g at $s_1 = 0$.

Proof. Note that $\mathbb{f}(1,1,1) = 2d - c$. If $2d = c$, then it holds that $\mathbb{F}(1^N) = 0^N$, whence $\mathbb{F}^t(1^N) = 0^N$, for any $t \geq 1$. If $2d \neq c$, then it holds that $2d - c \in (0,1)$ (since $(b,c,d) \in \mathcal{P}_0$). Taking into account that $g(s) = \mathbb{f}(s,s,s)$, it then follows that

$$\mathbb{F}^t(1^N) = (g^{t-1}(2d-c))^N \xrightarrow{t\to\infty} \lambda^N \ ,$$

according to property (g2) of Lemma 5.6.1. □

Since the behavior of $\mathbb{F}$ on 1^N is now understood, in the remainder we will only consider initial configurations $\boldsymbol{x}$ from $X_N^{0,1}$.

Lemma 5.6.3 *Let $\mathbb{f}$ be the local rule parameterized by a point $(b,c,d) \in \mathcal{P}_0$ and let $\mathbb{F}$ be the corresponding global rule. For each $\epsilon > 0$, for each $t_0 \in \mathbb{N}$ and for any initial configuration $\boldsymbol{x} \in X_N$, there exists $t_\epsilon \geq t_0$ such that*

$$\min\left(\mathbb{F}^{t_\epsilon}(\boldsymbol{x})\right) < \lambda + \epsilon \ .$$

Proof. We give a proof by contradiction. Let $\epsilon > 0$ and $t_0 \in \mathbb{N}$ be given and assume that for some initial configuration $\boldsymbol{x}$ it holds for each $t \geq t_0$ that

$$\min\left(\mathbb{F}^t(\boldsymbol{x})\right) \geq \lambda + \epsilon \ . \tag{5.4}$$

For every $t \geq t_0$ and any $n \in \{0,1,\ldots,N-1\}$, it holds that

$$\begin{aligned} x_n^{t+1} &= \mathbb{f}(x_{n-1}^t, x_n^t, x_{n+1}^t) \\ &= -(b+c-d)(x_{n-1}^t + x_{n+1}^t)x_n^t + b(x_{n-1}^t + x_{n+1}^t) + cx_n^t \\ &\leq -(b+c-d)(x_{n-1}^t + x_{n+1}^t)(\lambda+\epsilon) + b(x_{n-1}^t + x_{n+1}^t) + cx_n^t \\ &= \left(\frac{1}{2} - \frac{1}{2}c - (b+c-d)\epsilon\right)\left(x_{n-1}^t + x_{n+1}^t\right) + cx_n^t \ , \end{aligned}$$

since $b + c - d > 0$, $x_n^t \geq \lambda + \epsilon$ and $\lambda = \frac{2b+c-1}{2(b+c-d)}$. Hence, for each $t \geq t_0$ we find

$$\begin{aligned} 0 &\leq \sigma(\mathbb{F}^{t+1}(\boldsymbol{x})) \\ &\leq \sum_{n=0}^{N-1}\left[\left(\frac{1}{2} - \frac{1}{2}c - (b+c-d)\epsilon\right)(x_{n-1}^t + x_{n+1}^t) + cx_n^t\right] \\ &= (1 - 2(b+c-d)\epsilon)\,\sigma(\mathbb{F}^t(\boldsymbol{x})) \ . \end{aligned}$$

Note that $1 - 2(b + c - d)\epsilon < 1$, so the above inequality implies

$$\sigma(\mathbb{F}^t(\boldsymbol{x})) \xrightarrow{t\to\infty} 0 \ ,$$

contrary to the assumption in Eq. (5.4). □

Based on Theorem 3.1.1 let us define the set

$$\mathcal{V} = \{\mathbb{f}(x, y, z) \mid x, y, z \in \{m, M\}\} \ .$$

Note that if $\mathbb{f}$ is a local rule defined by Eq. (5.1), then the set $\mathcal{V}$ contains at most six different numbers:

$$\begin{aligned}
V_1 &= \mathbb{f}(m, m, m) = -2(b + c - d)m^2 + (2b + c)m \ ; \\
V_2 &= \mathbb{f}(M, m, m) = -(b + c - d)Mm - (b + c - d)m^2 + bM + (b + c)m \ ; \\
V_3 &= \mathbb{f}(m, M, m) = -2(b + c - d)Mm + 2bm + cM \ ; \\
V_4 &= \mathbb{f}(M, m, M) = -2(b + c - d)Mm + 2bM + cm \ ; \\
V_5 &= \mathbb{f}(M, M, m) = -(b + c - d)Mm - (b + c - d)M^2 + bm + (b + c)M \ ; \\
V_6 &= \mathbb{f}(M, M, M) = -2(b + c - d)M^2 + (2b + c)M \ .
\end{aligned}$$

Moreover, V_2 always lies between V_1 and V_4, while V_5 always lies between V_3 and V_6 (since $\mathbb{f}$ is affine also w.r.t. $x + z$). The following lemma presents some dependencies between V_1, V_3, V_4 and V_6.

Lemma 5.6.4 *Let $\mathbb{f}$ be the local rule parameterized by a point $(b, c, d) \in \mathcal{P}_0$ and let $0 \le m < M \le 1$. Then it holds that*

(d1) $V_1 \le V_4$ *if and only if* $m \le \frac{b}{b+c-d}$.
(d2) $V_3 \le V_6$ *if and only if* $M \le \frac{b}{b+c-d}$.
(d3) $V_1 \le V_3$ *if and only if* $m \le \frac{c}{2(b+c-d)}$.
(d4) $V_4 \le V_3$ *if and only if* $c \ge 2b$.
(d5) $V_4 \le V_6$ *if and only if* $M \le \frac{c}{2(b+c-d)}$.

Proof. Statements (d1)–(d5) respectively follow from

$$\begin{aligned}
V_4 - V_1 &= 2(M - m)(b - (b + c - d)m) \ , \\
V_6 - V_3 &= 2(M - m)(b - (b + c - d)M) \ , \\
V_3 - V_1 &= (M - m)(c - 2(b + c - d)) \ , \\
V_3 - V_4 &= (M - m)(c - 2b) \ , \\
V_4 - V_6 &= (M - m)\,(2(b + c - d)M - c) \ ,
\end{aligned}$$

which concludes the proof. □

Equipped with the above facts, we study the behavior of ACCAs parameterized by points in $\mathcal{P}_0$. We divide $\mathcal{P}_0$ into several parts, because for each part we use different proof methods.

5.6.2 The case $(b, c, d) \in \mathcal{P}_0 \setminus (P_1P_3 \cup P_4P_6 \cup P_5P_7)$.

In this subsection, we uncover the dynamics of the local rules parameterized by points (b, c, d) that belong to $\mathcal{P}_0$ but do not lie on any of the

edges P_1P_3, P_4P_6 and P_5P_7, *i.e.*, $0 < b < \frac{1}{2}$ or $0 < c < 1$ and presented in Figure 5.8. In this case, the dynamics turns out to be very simple: each

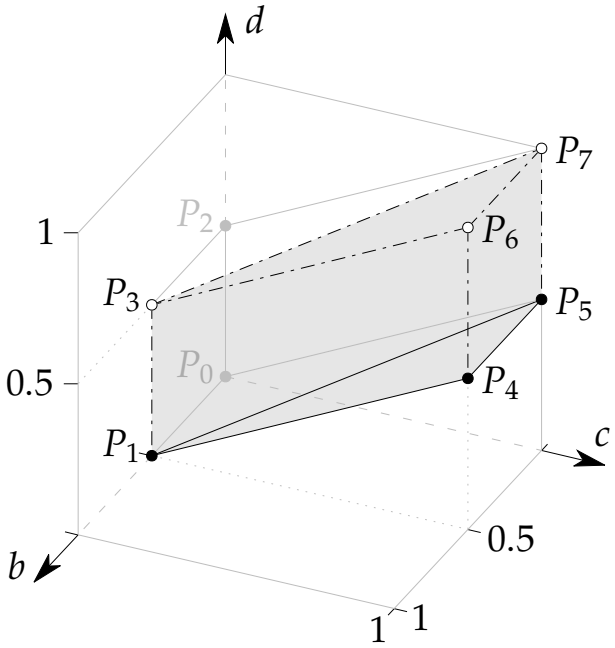


Figure 5.8: The polyhedron $\mathcal{P}_0 \setminus (P_1P_3 \cup P_4P_6 \cup P_5P_7)$.

initial configuration $\boldsymbol{x} \in X_N^{0,1}$ tends to the same homogeneous one, as stated in the following theorem.

Theorem 5.6.5 *Let* $\mathbb{f}$ *be the local rule parameterized by a point* $(b, c, d) \in \mathcal{P}_0$ *and let* $\mathbb{F}$ *be the corresponding global rule. If* $0 < b < \frac{1}{2}$ *or* $0 < c < 1$, *then for any initial configuration* $\boldsymbol{x} \in X_N^{0,1}$, *it holds that*

$$\lim_{t\to\infty} \mathbb{F}^t(\boldsymbol{x}) = \lambda^N .$$

Proof. Let $(b, c, d) \in \mathcal{P}_0$, where $0 < b < \frac{1}{2}$ or $0 < c < 1$, and $\boldsymbol{x} \in X_N^{0,1}$. We consider the case $c \geq 2b$, the case $c < 2b$ being analogous. Suppose that $b = \frac{1}{2}$, then it would also hold that $c = 1$, a contradiction. Hence, it holds that $b < \frac{1}{2}$, which implies that $\epsilon = \frac{1-2b}{2(b+c-d)} > 0$. Since $2d - c < 1$, according to Lemma 5.3.1(p2)–(p3) and Lemma 5.6.3, there exists $t_\epsilon \geq 2$ such that $0 < m_0 < \lambda + \epsilon$ and $M_0 < 1$, where

$$m_0 = \min(\mathbb{F}^{t_\epsilon}(\boldsymbol{x})) \text{ and } M_0 = \max(\mathbb{F}^{t_\epsilon}(\boldsymbol{x})) .$$

If $m_0 = M_0$, *i.e.*, $\mathbb{F}^{t_\epsilon}(\boldsymbol{x})$ is a homogeneous configuration $(m_0)^N$, then according to Lemma 5.6.1, it holds that $\mathbb{F}^t(\boldsymbol{x})$ tends to λ^N. Therefore, we can assume that $m_0 < M_0$.

Consider the following two sequences $(m_t)_{t\geq 0}$ and $(M_t)_{t\geq 0}$:

$$\begin{aligned} m_{t+1} &= \min\{\mathbb{f}(x, y, z) \mid x, y, z \in \{m_t, M_t\}\} , \\ M_{t+1} &= \max\{\mathbb{f}(x, y, z) \mid x, y, z \in \{m_t, M_t\}\} . \end{aligned}$$

According to Theorem 3.1.1, it holds that

$$\mathbb{F}^{t_\epsilon+t}(\boldsymbol{x}) \in [m_t, M_t]^N .$$

We will show that the sequence of N-dimensional cubes $[m_t, M_t]^N$ collapses to a single point λ^N. We consider two cases.

(i) *Assume that for some $t_0 \geq 0$, it holds that $M_{t_0} \leq \frac{b}{b+c-d}$.* Then obviously $m_{t_0} \leq \frac{b}{b+c-d}$ and from Lemma 5.6.4 it follows that

$$M_{t_0+1} = \mathbb{f}(M_{t_0}, M_{t_0}, M_{t_0}) = g(M_{t_0})$$

and

$$m_{t_0+1} = \mathbb{f}(m_{t_0}, m_{t_0}, m_{t_0}) = g(m_{t_0}) .$$

Note that $c \geq 2b$ implies that $\lambda \leq \frac{b}{b+c-d} \leq s_v$. Thus,

$$M_{t_0+1} = g(M_{t_0}) \leq g\left(\frac{b}{b+c-d}\right) \leq \frac{b}{b+c-d} .$$

By simple induction we get $M_t \leq \frac{b}{b+c-d}$ for $t \geq t_0$ and consequently $M_{t+1} = g(M_t)$ and $m_{t+1} = g(m_t)$. Therefore, according to Lemma 5.6.1(g2), we have

$$\lim_{t\to\infty} M_t = \lim_{t\to\infty} g(M_t) = \lim_{t\to\infty} g^t(M_{t_0}) = \lambda$$

and

$$\lim_{t\to\infty} m_t = \lim_{t\to\infty} g(m_t) = \lim_{t\to\infty} g^t(m_{t_0}) = \lambda ,$$

which concludes the proof in the first case.

(ii) *Assume that for all $t \geq 0$, it holds that $M_t > \frac{b}{b+c-d}$.* Let us note that for all $t \geq 0$ it holds that $m_t < \lambda + \epsilon$. Indeed, from Lemma 5.6.1(g1), it follows that

$$m_{t+1} \leq \mathbb{f}(m_t, m_t, m_t) = g(m_t) < \lambda + \epsilon = \frac{c}{2(b+c-d)} .$$

It follows from Lemma 5.6.4(d2), (d3) and (d4) that for all $t \geq 0$ it holds that

$$M_{t+1} = \mathbb{f}(m_t, M_t, m_t) = cM_t + 2m_t(b - (b+c-d)M_t) \leq cM_t , \quad (5.5)$$

which is only possible for $c = 1$, as otherwise the sequence $(M_t)_{t=0}^{\infty}$ would decrease exponentially to zero, contradicting the assumption that $M_t > \frac{b}{b+c-d} > 0$, for all $t \geq 0$. Note that if $c = 1$, then $\frac{b}{b+c-d} = \lambda < M_t$ and the sequence $(M_t)_{t=0}^{\infty}$ is decreasing. Again, we consider two cases.

(a) *Assume that for all $t \geq 0$, it holds that also $m_t \geq \lambda$.* It then holds that $\lambda \leq m_t \leq M_t$. Using Eq. (5.5), it follows that

$$\begin{aligned} M_{t+1} - \lambda &= M_t + 2m_t(b+1-d)(\lambda - M_t) - \lambda \\ &= (M_t - \lambda)(1 - 2m_t(b+1-d)) \\ &\leq (1-2b)(M_t - \lambda) . \end{aligned}$$

Since $0 < b < \frac{1}{2}$, the above upper bound shows that the difference between M_t and λ decreases exponentially to 0, *i.e.*, $\lim_{t\to\infty} M_t = \lambda$, which implies that also $\lim_{t\to\infty} m_t = \lambda$.

(b) *Assume that for some $\tau \geq 0$, it holds that $m_\tau < \lambda$.* Of course, then due to Lemma 5.6.1(g1), it holds that $m_t \leq \lambda$ for all $t \geq \tau$. It follows from Lemma 5.6.4(d1) and (d2) that for all $t \geq \tau$, it

holds that

$$m_{t+1} = \min(g(m_t), g(M_t)) \ . \tag{5.6}$$

Let $\delta = \min(m_\tau, g(M_\tau))$. On the one hand, it holds that $0 < m_\tau \leq \lambda$. On the other hand, $\lambda < M_\tau < 1$ implies $g(M_\tau) > 0$. This yields $0 < \delta \leq \lambda$. We will show that $m_t \geq \delta$ for all $t \geq \tau$ (it holds for $t = \tau$). The following reasoning is based on Figure 5.6a. Since $\lambda < M_t \leq M_\tau$, it holds that $g(M_t) \geq \min(\lambda, g(M_\tau)) \geq \delta$. Moreover, since $m_t \leq \lambda$, it holds that $g(m_t) \geq m_t \geq \delta$. Together, according to Eq. (5.6), we obtain $m_{t+1} \geq \delta$.

Now, let us compare the differences $M_{t+1} - m_{t+1}$ and $M_t - m_t$ for $t \geq \tau$. If $m_{t+1} = g(M_t)$, then it follows from Eq. (5.5) and Lemma 5.6.1 that

$$\begin{aligned} M_{t+1} - m_{t+1} &= (M_t - m_t)(2(b + 1 - d)M_t - 2b) \\ &\leq (M_t - m_t)2(1 - d)M_t \\ &\leq (M_t - m_t)2(1 - d)M_\tau \ . \end{aligned}$$

Similarly, if $m_{t+1} = g(m_t)$, then

$$\begin{aligned} M_{t+1} - m_{t+1} &= (M_t - m_t)(1 - 2(b + 1 - d)m_t) \\ &\leq (M_t - m_t)(1 - 2(b + 1 - d)\delta) \ . \end{aligned}$$

In both cases, we have

$$M_{t+1} - m_{t+1} \leq \alpha(M_t - m_t)$$

for all $t \geq \tau$, where

$$\alpha = \max\left(2(1 - d)M_\tau, 1 - 2(b + 1 - d)\delta\right) \ ,$$

Note that $c = 1$ implies that $\frac{1}{2} \leq d < 1$. Together with $\delta > 0$ and $b < \frac{1}{2}$, it then follows that $\alpha < 1$. This means hat

$$\lim_{t\to\infty} m_t = \lim_{t\to\infty} M_t = \lambda \ ,$$

as $m_t \leq \lambda \leq M_t$, for all $t \geq \tau$.

□

5.6.3 The case $(b, c, d) = (0.5, 1, d)$ for $0.5 \leq d < 1$.

In this subsection, we discuss the dynamics of the local rules parameterized by points belonging to the edge P_4P_6 as presented in Figure 5.9. In this case, $\lambda = \frac{1}{3-2d}$ and the local rule is given by

$$\mathbb{f}(x, y, z) = -\left(\frac{3}{2} - d\right)(x + z)y + \frac{1}{2}(x + z) + y \ .$$

An interesting observation is the following bifurcation: for any even

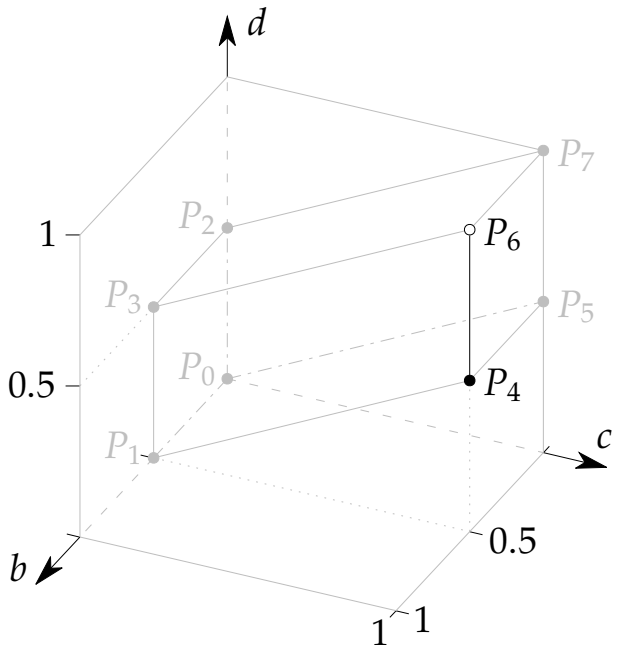


Figure 5.9: The edge P_4P_6 in parallelepiped $\mathcal{P}$.

$N \in \mathbb{N}$, it holds that

$$\lim_{t\to\infty} \mathbb{F}^t\left((01)^{\frac{N}{2}}\right) = \lambda^N \text{ if } \frac{1}{2} < d < 1, \text{ and}$$

$\lim_{t\to\infty} \mathbb{F}^t\left((01)^{\frac{N}{2}}\right) = 0^N$ if $d = \frac{1}{2}$, while

$$\lim_{d\to\frac{1}{2}^+} \lambda = \lim_{d\to\frac{1}{2}^+} \frac{1}{3-2d} = \frac{1}{2} \neq 0\,.$$

Theorem 5.6.6 *Let $\mathbb{f}$ be the local rule parameterized by a point $\left(\frac{1}{2}, 1, d\right) \in \mathcal{P}_0$ and let $\mathbb{F}$ be the corresponding global rule. If $\frac{1}{2} < d < 1$, then for any initial configuration $\boldsymbol{x} \in X_N^{0,1}$, it holds that*

$$\lim_{t\to\infty} \mathbb{F}^t(\boldsymbol{x}) = \lambda^N\,.$$

If $d = \frac{1}{2}$, then

- *if N is odd, then for any $\boldsymbol{x} \in X_N^{0,1}$, it holds that*

$$\lim_{t\to\infty} \mathbb{F}^t(\boldsymbol{x}) = \lambda^N = \left(\frac{1}{2}\right)^N\,;$$

- *if N is even, then for any $\boldsymbol{x} \in X_N^{0,1} \setminus \left\{(01)^{\frac{N}{2}}, (10)^{\frac{N}{2}}\right\}$, it holds that*

$$\lim_{t\to\infty} \mathbb{F}^t(\boldsymbol{x}) = \lambda^N = \left(\frac{1}{2}\right)^N\,,$$

while $\mathbb{F}^t\left((01)^{\frac{N}{2}}\right) = \mathbb{F}^t\left((10)^{\frac{N}{2}}\right) = 0^N$ for $t \geq 2$.

Proof. Let $\boldsymbol{x} \in X_N^{0,1}$. For any $t \geq 0$, let μ_t denote the maximal absolute

difference between x_n^t and λ, *i.e.*, $\mu_t = \max_{0 \leq n \leq N-1} \left| x_n^t - \lambda \right|$. As $\frac{1}{2} \leq \lambda < 1$, it holds that $\mu_t \leq \lambda$. Additionally,

$$\mathbb{f}(x, y, z) - \lambda = (\lambda - y)\left(\frac{3}{2} - d\right)(x - \lambda + z - \lambda)$$

implies

$$|\mathbb{f}(x, y, z) - \lambda| \leq |y - \lambda|\left(\frac{3}{2} - d\right)(|x - \lambda| + |z - \lambda|) \ ,$$

which means that

$$\mu_{t+1} \leq \mu_t \left(\frac{3}{2} - d\right) 2\mu_t = (3 - 2d)\mu_t^2 \ . \tag{5.7}$$

In particular, since $\mu_t \leq \lambda = \frac{1}{3-2d}$, it holds that $\mu_{t+1} \leq \mu_t$.

Lemma 5.3.1(p2) implies that there exists $\tau \geq 2$ such that

$$m = \min\left(\mathbb{F}^\tau(\boldsymbol{x})\right) > 0 \, .$$

Similarly, Lemma 5.3.1(p3) implies that there exists some $\tau > \frac{N}{2} + 2$ such that

$$M = \max\left(\mathbb{F}^\tau(\boldsymbol{x})\right) < 1 \ ,$$

in the following cases: (i) $\frac{1}{2} < d < 1$, (ii) $d = \frac{1}{2}$ and N is odd, or (iii) $d = \frac{1}{2}$, N is even and $\boldsymbol{x} \in X_N^{0,1} \setminus \left\{(01)^{\frac{N}{2}}, (10)^{\frac{N}{2}}\right\}$. Hence, there exists some $\tau > 0$ such that

$$m = \min\left(\mathbb{F}^\tau(\boldsymbol{x})\right) > 0 \quad \text{and} \quad M = \max\left(\mathbb{F}^\tau(\boldsymbol{x})\right) < 1 \, .$$

This implies that μ_τ is strictly smaller than λ. Indeed,

$$\mu_\tau = \max_{0 \leq n \leq N-1} |x_n^\tau - \lambda| = \max\left(|M - \lambda|, |m - \lambda|\right) < \lambda \ .$$

Therefore, according to Eq. (5.7), for any $t \geq \tau$ it holds that

$$\mu_{t+1} \leq \Big((3 - 2d)\mu_\tau\Big)\mu_t \ ,$$

where $(3 - 2d)\mu_\tau < 1$. As a consequence, we get that $\mu_t \xrightarrow{t \to \infty} 0$, which means that $\lim_{t \to \infty} \mathbb{F}^t(\boldsymbol{x}) = \lambda^N$.

Note that if $d = \frac{1}{2}$, then $\lambda = \frac{1}{2}$ and

$$\mathbb{f}(x, y, z) = -(x + z)y + \frac{1}{2}(x + z) + y \, .$$

One easily verifies that in this case

$$\mathbb{F}^t\left((01)^{\frac{N}{2}}\right) = \mathbb{F}^t\left((10)^{\frac{N}{2}}\right) = 0^N$$

for each $t \geq 2$. □

5.6.4 The case $(b, c, d) = (0.5, 0, d)$ for $0 \leq d < 0.5$.

In this subsection, we discuss the dynamics of the local rules parameterized by points belonging to the edge P_1P_3 as presented in Figure 5.10. In

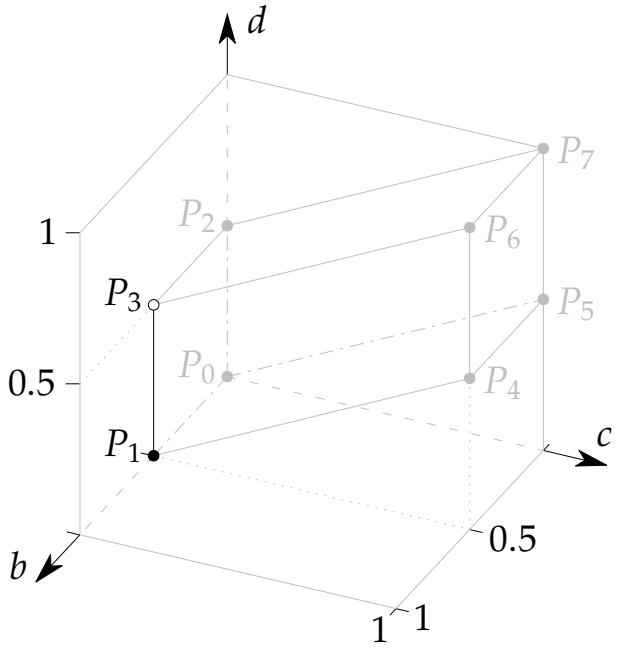


Figure 5.10: The edge P_1P_3 in parallelepiped $\mathcal{P}$.

this case, the local rule is given by

$$\mathbb{f}(x, y, z) = \frac{1}{2}(x + z)\,(1 - (1 - 2d)\,y)\ , \tag{5.8}$$

which yields

$$\mathbb{f}(x, y, z) \leq \frac{1}{2}(x + z) \leq \max(x, z)\ . \tag{5.9}$$

Moreover, the alternating sum of states, *i.e.*,

$$(-1)^t x_0^t + (-1)^{t+1} x_1^t + (-1)^{t+2} x_2^t + \ldots + (-1)^{t+N-1} x_{N-1}^t\ ,$$

is an invariant of this dynamical system. Indeed,

$$\sum_{j=0}^{N-1} (-1)^{t+1+j} x_j^{t+1} = \sum_{j=0}^{N-1} (-1)^{t+1+j} \left[\frac{1}{2} x_{j-1}^t + \frac{1}{2} x_{j+1}^t + \right.$$
$$\left. - \frac{1}{2}\,(1 - 2d) \left(x_{j-1}^t x_j^t + x_j^t x_{j+1}^t \right) \right]\ .$$

Due to the periodic boundary conditions, we have

$$\sum_{j=0}^{N-1} (-1)^j x_{j-1}^t = \sum_{j=0}^{N-1} (-1)^j x_{j+1}^t = \sum_{j=0}^{N-1} (-1)^{j+1} x_j^t$$

and

$$\sum_{j=0}^{N-1} (-1)^j x_{j-1}^t x_j^t = \sum_{j=0}^{N-1} (-1)^{j+1} x_j^t x_{j+1}^t = - \sum_{j=0}^{N-1} (-1)^j x_j^t x_{j+1}^t\ ,$$

which implies

$$\sum_{j=0}^{N-1}(-1)^{t+1+j}x_j^{t+1} = \sum_{j=0}^{N-1}(-1)^{t+j}x_j^t \,. \tag{5.10}$$

The following theorem describes the dynamics of such CAs.

Theorem 5.6.7 *Let* $\mathbb{f}$ *be the local rule parameterized by a point* $(\frac{1}{2}, 0, d) \in \mathcal{P}_0$, *where* $0 \leq d < \frac{1}{2}$, *and let* $\mathbb{F}$ *be the corresponding global rule.*

(i) If N *is even, then for any* $\boldsymbol{x} \in X_N$, *it holds that*

$$\lim_{t\to\infty}\mathbb{F}^{2t}(\boldsymbol{x}) = (\alpha 0)^{\frac{N}{2}} \text{ and } \lim_{t\to\infty}\mathbb{F}^{2t+1}(\boldsymbol{x}) = (0\alpha)^{\frac{N}{2}}$$

if $\alpha = \frac{2}{N}(x_0 - x_1 + x_2 - x_3 + \ldots - x_{N-1}) \geq 0$, *while*

$$\lim_{t\to\infty}\mathbb{F}^{2t}(\boldsymbol{x}) = (0|\alpha|)^{\frac{N}{2}} \text{ and } \lim_{t\to\infty}\mathbb{F}^{2t+1}(\boldsymbol{x}) = (|\alpha|0)^{\frac{N}{2}}$$

if $\alpha < 0$.

(ii) If N *is odd, then for any* $\boldsymbol{x} \in X_N$, *it holds that*

$$\lim_{t\to\infty}\mathbb{F}^{t}(\boldsymbol{x}) = 0^N \,.$$

Proof. Let $\boldsymbol{x} \in X_N$. First, we consider even N. For each $t \geq 0$, we consider the following two sets

$$\begin{aligned} E_t &= \left\{ j \in \{0, 1, \ldots, N-1\} \mid j \equiv t \pmod 2 \right\} \\ O_t &= \left\{ j \in \{0, 1, \ldots, N-1\} \mid j \not\equiv t \pmod 2 \right\} \\ &= \{0, 1, \ldots, N-1\} \setminus E_t \,. \end{aligned}$$

Note that

$$\begin{aligned} E_0 = E_2 = E_4 = \ldots &= \{0, 2, \ldots, N-2\} \,, \\ E_1 = E_3 = E_5 = \ldots &= \{1, 3, \ldots, N-1\} \,, \\ O_0 = O_2 = O_4 = \ldots &= \{1, 3, \ldots, N-1\} \,, \\ O_1 = O_3 = O_5 = \ldots &= \{0, 2, \ldots, N-2\} \,. \end{aligned}$$

Let $\alpha_t = \max_{j\in E_t} x_j^t$ and $\beta_t = \max_{j\in O_t} x_j^t$. According to Eq. (5.9), for any $t \geq 0$, it holds that $\alpha_{t+1} \leq \alpha_t$ and $\beta_{t+1} \leq \beta_t$. This means that both sequences $(\alpha_t)_{t\geq 0}$ and $(\beta_t)_{t\geq 0}$ are convergent. Hence, $\alpha = \lim_{t\to\infty}\alpha_t$ and $\beta = \lim_{t\to\infty}\beta_t$. Now consider $\delta > 0$, then there exists $\tau \geq 0$ such that for any $t \geq \tau$, it holds that $\alpha \leq \alpha_t < \alpha + \delta$. We will show that for any $t \geq \tau$ and any $j \in E_t$, it holds that

$$x_j^t \geq \alpha - \left(2^{\frac{N}{2}-1} - 1\right)\delta \,. \tag{5.11}$$

Indeed, suppose that this is not true, *i.e.*, there exist $t_0 \geq \tau$ and $j_0 \in E_{t_0}$ such that $x_{j_0}^{t_0} < \alpha - \epsilon$, where $\epsilon = \left(2^{\frac{N}{2}-1} - 1\right)\delta$. According to Eq. (5.9) we have

$$\begin{aligned} x_{j_0-1}^{t_0+1}, x_{j_0+1}^{t_0+1} &< \frac{1}{2}(\alpha + \delta + \alpha - \epsilon) = \alpha - \frac{1}{2}\epsilon + \frac{1}{2}\delta \\ x_{j_0-2}^{t_0+2}, x_{j_0}^{t_0+2}, x_{j_0+2}^{t_0+2} &< \frac{1}{2}(\alpha + \delta + \alpha - \frac{1}{2}\epsilon + \frac{1}{2}\delta) = \alpha - \frac{1}{4}\epsilon + \frac{3}{4}\delta \end{aligned}$$

and so on. Thus, after $\frac{N}{2} - 1$ steps, we get that for each $j \in E_{t_0+\frac{N}{2}-1}$

$$x_j^{t_0+\frac{N}{2}-1} < \alpha - \frac{1}{2^{\frac{N}{2}-1}}\epsilon + \frac{2^{\frac{N}{2}-1}-1}{2^{\frac{N}{2}-1}}\delta = \alpha \ ,$$

which contradicts the fact that $\alpha_{t_0+k-1} \geq \alpha$, which implies

$$\min_{j\in E_t} x_j^t \xrightarrow{t\to\infty} \alpha \ .$$

In the same way, we can prove that $\min_{j\in O_t} x_j^t \xrightarrow{t\to\infty} \beta$. It then holds that

$$\lim_{t\to\infty} \mathbb{F}^{2t}(\boldsymbol{x}) = (\alpha\beta)^{\frac{N}{2}} \text{ and } \lim_{t\to\infty} \mathbb{F}^{2t+1}(\boldsymbol{x}) = (\beta\alpha)^{\frac{N}{2}} \ .$$

According to Eqs. (5.8) and (5.10), the numbers α and β satisfy the following equations:

$$\begin{cases} \alpha & = \alpha\,(1-(1-2d)\,\beta) \\ \beta & = \beta\,(1-(1-2d)\,\alpha) \\ \frac{N}{2}(\alpha-\beta) & = x_0 - x_1 + x_2 - x_3 + \ldots - x_{N-1} \ . \end{cases}$$

The first two imply that if $\alpha \neq 0$, then $\beta = 0$ and vice versa. Therefore, if $x_0 - x_1 + x_2 - x_3 + \ldots - x_{N-1} \geq 0$, then $\alpha = \frac{2}{N}(x_0 - x_1 + x_2 - x_3 + \ldots - x_{N-1})$ and $\beta = 0$. Otherwise, $\alpha = 0$ and $\beta = -\frac{2}{N}(x_0 - x_1 + x_2 - x_3 + \ldots - x_{N-1})$. This concludes the proof for even N.

Second, we consider odd N. Consider the following concatenation:

$$(\boldsymbol{xx}) = x_0x_1\ldots x_{N-1}x_0x_1\ldots x_{N-1} \in X_{2N} \ .$$

Due to the periodic boundary conditions, it holds for each $t \geq 0$ that

$$\mathbb{F}^t(\boldsymbol{xx}) = \mathbb{F}^t(\boldsymbol{x})\mathbb{F}^t(\boldsymbol{x}) \ .$$

However, for the initial configuration $\boldsymbol{xx}$, the alternating sum equals 0 and the first part of the theorem implies that

$$\lim_{t\to\infty} \mathbb{F}^{2t}(\boldsymbol{xx}) = \lim_{t\to\infty} \mathbb{F}^{2t+1}(\boldsymbol{xx}) = 0^{2N} \ .$$

This concludes the proof for odd N. □

5.6.5 The case $(b, c, d) = (0, 1, d)$ for $0.5 \leq d < 1$.

In this subsection, we discuss the dynamics of the local rules parameterized by points belonging to the edge P_5P_7 as presented in Figure 5.11. In this case, $\lambda = 0$ and the local rule is given by

$$\mathbb{f}(x, y, z) = y(1 - (1 - d)(x + z)) \ .$$

In particular, $\mathbb{f}(x, 0, z) = 0$ and $\mathbb{f}(0, y, 0) = y$. This means that any binary configuration $\boldsymbol{x} \in X_N$ not containing two neighboring 1s is a fixed point of $\mathbb{f}$. Furthermore, each substring of $\boldsymbol{x} \in X_N$ starting and ending with 0 and not containing two neighboring 1s remains unchanged in

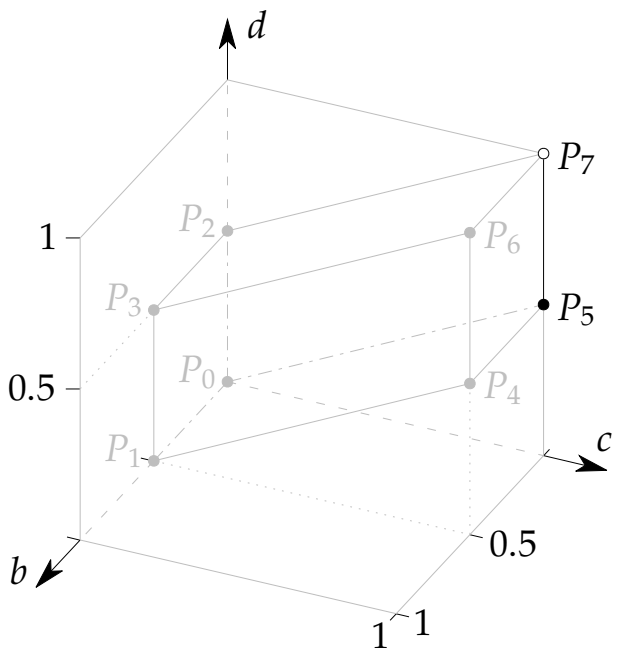


Figure 5.11: The edge P_5P_7 in parallelepiped $\mathcal{P}$.

$\mathbb{F}^t(\boldsymbol{x})$ for every $t \geq 0$. Additionally, according to Lemma 5.6.2, we have $\mathbb{F}^t(1^N) \xrightarrow{t\to\infty} 0^N$.

For these reasons, to uncover the behavior of $\mathbb{f}$, it suffices to consider strings of the type $011\ldots110$, where the number of 1s equals K with $2 \leq K < N$. W.l.o.g., let us assume that

$$x_0^0 = 0, x_1^0 = x_2^0 = \ldots = x_K^0 = 1, x_{K+1}^0 = 0\ .$$

The dynamics of $\mathbb{f}$ on $x_0^0 x_1^0 \ldots x_K^0 x_{K+1}^0 = 011\ldots110$ is then described by the system of recurrence relations:

$$\begin{cases} x_0^{t+1} = x_{K+1}^{t+1} = 0 \\ x_n^{t+1} = x_n^t \left(1 - (1-d)(x_{n-1}^t + x_{n+1}^t)\right), n \in \{1, \ldots, K\}, \end{cases} \tag{5.12}$$

where $\frac{1}{2} \leq d < 1$. Since $0 \leq 1 - (1-d)(x_{n-1}^t + x_{n+1}^t) \leq 1$, the sequence $\left(x_n^t\right)_{t=0}^{\infty}$ is decreasing for every $n \in \{1, 2, \ldots, K\}$ and thus converges. We denote this limit as $q_n = \lim_{t\to\infty} x_n^t \in [0, 1]$. According to Eq. (5.12), it holds that

$$q_n = q_n \left(1 - (1-d)(q_{n-1} + q_{n+1})\right)\ , \tag{5.13}$$

for any $n \in \{1, 2, \ldots, K\}$, where $q_0 = q_{K+1} = 0$. Since $\mathbb{f}$ is outer-totalistic, it holds that

$$x_n^t = x_{K+1-n}^t \tag{5.14}$$

for any $n \in \{1, 2, \ldots, K\}$ and any $t \geq 0$, and thus

$$q_n = q_{K+1-n} \tag{5.15}$$

for any $n \in \{1, 2, \ldots, K\}$.

We start with some technical lemmata.

Lemma 5.6.8 *For each* $n \in \{1, 2, \ldots, K-1\}$, *it holds that* $q_n q_{n+1} = 0$.

Moreover, if K is even, then $q_{\frac{K}{2}} = q_{\frac{K}{2}+1} = 0$. In particular, if $K = 2$, then $q_1 = q_2 = 0$.

Proof. Since, $\frac{1}{2} \leq d < 1$, it follows from Eq. (5.13) that $q_{n-1}q_n + q_n q_{n+1} = 0$, for any $n \in \{1, 2, \ldots, K-1\}$. Since $q_0 = 0$, we get $q_1 q_2 = 0$. A simple induction yields the first part of the claim. If K is even, then, according to Eq. (5.15), it holds for $n = \frac{K}{2}$ that $q_{\frac{K}{2}} = q_{\frac{K}{2}+1}$. Hence, $q_{\frac{K}{2}} q_{\frac{K}{2}+1} = 0$ implies $q_{\frac{K}{2}} = q_{\frac{K}{2}+1} = 0$, which concludes the proof. □

The above lemma states that regardless of $d \in [\frac{1}{2}, 1]$, the string 0110 goes to 0000:

$$\ldots 0110 \ldots \xrightarrow{t\to\infty} \ldots 0000 \ldots \ .$$

From here on, we can assume that $K \geq 3$.

Lemma 5.6.9 *If $K \geq 3$, then $q_1 \geq 1 - d$. Moreover, if $d = \frac{1}{2}$, then the string $011 \ldots 110$ in $\boldsymbol{x}$ turns into the string $0\frac{1}{2}0 \ldots 0\frac{1}{2}0$ in $\mathbb{F}(\boldsymbol{x})$, which then remains unchanged in $\mathbb{F}^t(\boldsymbol{x})$ for any $t \geq 1$.*

Proof. According to Eq. (5.12), for any $t > 0$, it holds that

$$x_1^t - x_2^t = x_1^{t-1} - x_2^{t-1} + (1-d)x_2^{t-1}x_3^{t-1} \ .$$

Hence, also using $x_1^0 = x_2^0 = 1$, we have

$$x_1^t - x_2^t = x_1^0 - x_2^0 + (1-d)\sum_{l=0}^{t-1} x_2^l x_3^l = (1-d)\sum_{l=0}^{t-1} x_2^l x_3^l \ .$$

Since $x_2^0 x_3^0 = 1$ and $x_2^l x_3^l \geq 0$, for any $l \geq 0$, it then follows that

$$q_1 \geq q_1 - q_2 = (1-d)\sum_{l=0}^{\infty} x_2^l x_3^l \geq 1 - d \ .$$

Further, if $d = \frac{1}{2}$, then the local rule is given by $\mathbb{f}(x, y, z) = y(1 - \frac{1}{2}(x+z))$, and the second part of the lemma readily follows. □

Finally, we consider the case $\frac{1}{2} < d < 1$ and $K \geq 3$. The following theorem describes the dynamics of $\mathbb{F}$ on strings $011 \ldots 110$, where the number of 1s is even.

Theorem 5.6.10 *Let $\frac{1}{2} < d < 1$ and $K \geq 4$ be even. For any $t \geq 0$, it holds that*

$$x_1^t \geq x_3^t \geq \ldots \geq x_{K-3}^t \geq x_{K-1}^t \tag{5.16}$$

and, as a consequence,

$$q_1 \geq q_3 \geq \ldots \geq q_{K-3} \geq q_{K-1} \ . \tag{5.17}$$

Moreover, $q_{2j} = q_{K+1-2j} = 0$, for $2j \leq \frac{K}{2}$.

Proof. To prove Eq. (5.16), we apply induction w.r.t. t. For $t = 0$, these inequalities are obvious. Assume that they are satisfied for some $t \geq 0$. Using the symmetry relationships in Eq. (5.14), we have

$$x_{2j}^t \leq x_{2j+2}^t$$

for any $j \in \{1, 2, \ldots, \frac{K}{2} - 1\}$. Using the induction hypothesis, it then follows that

$$\begin{aligned} x_{2j-1}^{t+1} &= x_{2j-1}^t (1 - (1-d)(x_{2j-2}^t + x_{2j}^t)) \\ &\geq x_{2j+1}^t (1 - (1-d)(x_{2j}^t + x_{2j+2}^t)) = x_{2j+1}^{t+1} , \end{aligned}$$

for any $j \in \{1, 2, \ldots, \frac{K}{2} - 1\}$, which yields Eq. (5.16) and thus Eq. (5.17). According to Lemma 5.6.8, it holds that $q_{\frac{K}{2}} = q_{\frac{K}{2}+1} = 0$, so Eq. (5.17) implies that $q_{2j+1} = 0$, for any $2j + 1 \geq \frac{K}{2}$. Thus

$$q_{2j} = q_{K+1-2j} = 0,$$

for any $2j \leq \frac{K}{2}$, which concludes the proof.

□

The above theorem can be illustrated as follows. Suppose that in a string $011 \ldots 110$ the number of 1s is even, then

$$\begin{array}{cccccccccccccc} 0 & 1 & 1 & 1 & \ldots & 1 & \vdots & 1 & \ldots & 1 & 1 & 1 & 0 \\ & & & & & & \downarrow & t \to \infty & & & & & \\ 0 & q_1 & 0 & q_3 & \ldots & 0 & \vdots & 0 & \ldots & q_3 & 0 & q_1 & 0 \end{array}$$

where $q_1 \geq q_3 \geq \ldots$ and $q_1 \geq 1 - d$.

For an odd number $K \geq 3$, we introduce the following notations: $s = \frac{K+1}{2}$, $2n_o - 1$ is the largest odd number not greater than s and $2n_e$ is the largest even number not greater than s.

Lemma 5.6.11 *Let $K \geq 3$ be odd. For any $t \geq 0$, it holds that*

$$\begin{aligned} x_1^t \geq x_3^t \geq \ldots \geq x_{2n_o-1}^t \\ x_2^t \leq x_4^t \leq \ldots \leq x_{2n_e}^t . \end{aligned} \tag{5.18}$$

Additionally,

$$x_{2n_o-1}^t \geq x_{2n_e}^t . \tag{5.19}$$

Moreover, if $t > s$, then the inequalities in Eqs. (5.18) and (5.19) are strict.

Proof. One can prove Eq. (5.18) using induction in a similar way as in the proof of Theorem 5.6.10. We can also prove Eq. (5.19) using induction. Note that for $t = 0$ the inequality is trivially fulfilled and assume that it

is satisfied for some $t \geq 0$. If s is odd, then $2n_o - 1 = s$ and $2n_e = s - 1$, thus Eq. (5.18) and the induction hypothesis imply

$$\begin{aligned} x^{t+1}_{2n_e} = x^{t+1}_{s-1} &= x^t_{s-1}(1-(1-d)(x^t_{s-2}+x^t_s)) \\ &\leq x^t_s(1-(1-d)(x^t_s+x^t_s)) \\ &\leq x^t_s(1-(1-d)(x^t_{s-1}+x^t_{s-1})) \\ &= x^t_s(1-(1-d)(x^t_{s-1}+x^t_{s+1})) \\ &= x^{t+1}_s = x^{t+1}_{2n_o-1} \,. \end{aligned}$$

Similarly, if s is even, then $2n_e = s$ and $2n_o - 1 = s - 1$, and it follows using Eq. (5.18), the induction hypothesis and the fact that $x^t_{s-1} = x^t_{s+1}$ according to Eq. (5.14), that

$$\begin{aligned} x^{t+1}_{2n_e} = x^{t+1}_s &= x^t_s(1-(1-d)(x^t_{s-1}+x^t_{s+1})) \\ &\leq x^t_s(1-(1-d)(x^t_s+x^t_s)) \\ &\leq x^t_{s-1}(1-(1-d)(x^t_{s-2}+x^t_s)) = x^{t+1}_{s-1} \\ &= x^{t+1}_{2n_o-1} \,. \end{aligned}$$

Observe that if $2j+1 < \frac{K}{2}$ and $x^t_{2j-1} > x^t_{2j+1}$, or $x^t_{2j-2} < x^t_{2j}$, or $x^t_{2j} < x^t_{2j+2}$, then $x^{t+1}_{2j-1} > x^{t+1}_{2j+1}$ and, consequently, $x^{t+k}_{2j-1} > x^{t+k}_{2j+1}$ for $k \geq 1$. Analogously, if $2j+2 < \frac{K}{2}$ and $x^t_{2j} < x^t_{2j+2}$, or $x^t_{2j-1} > x^t_{2j+1}$, or $x^t_{2j+1} > x^t_{2j+3}$, then $x^{t+1}_{2j} < x^{t+1}_{2j+2}$ and, consequently, $x^{t+k}_{2j} < x^{t+k}_{2j+2}$ for $k \geq 1$. Since $x^1_1 = 1-(1-d) > 1-2(1-d) = x^1_3$, by simple induction one can obtain strict inequalities in Eq. (5.18) for $t \geq s$. This also implies that the inequalities in Eq. (5.19) are strict.

□

Lemma 5.6.12 *Let $K \geq 3$ be odd. For any $t \geq 0$, it holds that*

$$x^{t+1}_{2n_o-1} - x^{t+1}_{2n_e} \geq x^t_{2n_o-1} - x^t_{2n_e} \,. \tag{5.20}$$

Proof. For odd s, we have $x^t_{2n_e} = x^t_{2n_e+2}$, thus

$$\begin{aligned} x^{t+1}_{2n_o-1} - x^{t+1}_{2n_e} &= x^t_{2n_o-1}(1-(1-d)(x^t_{2n_e}+x^t_{2n_e})) \\ &\quad - x^t_{2n_e}(1-(1-d)(x^t_{s-2}+x^t_{2n_o-1})) \\ &= x^t_{2n_o-1} - x^t_{2n_e} - (1-d)x^t_{2n_e}(x^t_{2n_o-1} - x^t_{s-2}) \\ &\geq x^t_{2n_o-1} - x^t_{2n_e} \,. \end{aligned}$$

For even s, we have $x^t_{2n_o-1} = x^t_{2n_o+1}$, thus

$$\begin{aligned} x^{t+1}_{2n_o-1} - x^{t+1}_{2n_e} &= x^t_{2n_o-1} - x^t_{2n_e} - (1-d)x^t_{2n_o-1}(x^t_{s-2} - x^t_{2n_e}) \\ &\geq x^t_{2n_o-1} - x^t_{2n_e} \,. \end{aligned}$$

□

Theorem 5.6.13 *Let $K \geq 3$ be odd. For any $2j-1 \leq s$, it holds that $q_{2j-1} > 0$, while for any $2j \leq s$, it holds that $q_{2j} = 0$. Moreover, $q_1 + q_3 + \ldots + q_K = 1$.*

Proof. If $t > s$, then for $2j - 1 \leq s$ Lemmata 5.6.11 and 5.6.12 imply that

$$x^t_{2j-1} \geq x^t_{2n_o-1} \geq x^t_{2n_o-1} - x^t_{2n_e} \geq x^{s+1}_{2n_o-1} - x^{s+1}_{2n_e} \ .$$

Thus $q_{2j-1} \geq x^{s+1}_{2n_o-1} - x^{s+1}_{2n_e} > 0$, since for $t = s + 1$ inequality Eq. (5.19) is strict. Furthermore, since $q_{2j-1} > 0$, Lemma 5.6.8 implies that $q_{2j} = 0$. To prove the last part of the theorem, note that since $x^t_0 = x^t_{K+1} = 0$, it holds that

$$\sum_{j=1}^{K}(-1)^{j+1}x^t_j = \sum_{j=1}^{K}(-1)^{j+1}x^{t-1}_j = \ldots = \sum_{j=1}^{K}(-1)^{j+1}x^0_j = 1 \ .$$

This implies that $\sum_{j=1}^{K}(-1)^{j+1}q_j = 1$. Since $q_j = 0$ for even j, the proof is complete.

□

The above theorem can be illustrated as follows. For even s, we have

$$\begin{array}{cccccccccccccccccc}
 & & & & & & & & & s & & & & & & & & \\
0 & 1 & 1 & 1 & 1 & \ldots & & 1 & & 1 & & 1 & & \ldots & 1 & 1 & 1 & 1 & 0 \\
 & & & & & & & & & \downarrow & t \to \infty & & & & & & & & \\
0 & q_1 & 0 & q_3 & 0 & \ldots & q_{2n_o-1} & & 0 & q_{2n_o-1} & & \ldots & 0 & q_3 & 0 & q_1 & 0 & &
\end{array}$$

while for odd s, we have

$$\begin{array}{cccccccccccccccccc}
 & & & & & & & & & s & & & & & & & & \\
0 & 1 & 1 & 1 & 1 & \ldots & 1 & & & 1 & & 1 & \ldots & 1 & 1 & 1 & 1 & 0 & \\
 & & & & & & & & & \downarrow & & t \to \infty & & & & & & & \\
0 & q_1 & 0 & q_3 & 0 & \ldots & 0 & q_{2n_o-1} & & 0 & \ldots & 0 & q_3 & 0 & q_1 & 0 & & &
\end{array}$$

5.7 Conclusions

In this chapter, we provided a complete description of the dynamics of legal outer-totalistic ACCAs. On the one hand, such CAs are the simplest generalization of ECAs, while on the other hand, they are dynamical systems that exhibit some properties that do not occur in the binary case. Thanks to massive numerical simulations, we have been able to partition the rule space in a number of classes with a distinct behavior. Through the use of a panoply of proof techniques and oftentimes tedious proofs, we have been able to provide undeniable analytical evidence for the dynamics observed.

Table 5.3 summarizes the results obtained. In order to keep the presentation manageable, we focus on initial configurations from $X^{0,1}_N$ only and add a short explanation for the other configurations separately below the table.

Table 5.3: Classes of ACCA rules with distinct dynamical behavior.

Conditions				Dynamics for $\boldsymbol{x} \in X_N^{0,1}$
$2d - c = 1$	$b + d < 1$			$\mathbb{F}^t(\boldsymbol{x}) \to 0^N$
	$b + d > 1$			$\mathbb{F}^t(\boldsymbol{x}) \to 1^N$
	$b + d = 1$	$0 < b < \frac{1}{2}$		$\mathbb{F}^t(\boldsymbol{x}) \to (\rho(\boldsymbol{x}))^N$
		$b = 0$		$\mathbb{F}^t(\boldsymbol{x}) = \boldsymbol{x}$
		$b = \frac{1}{2}$	N odd	$\mathbb{F}^t(\boldsymbol{x}) \to (\rho(\boldsymbol{x}))^N$
			N even	$\mathbb{F}^{2t}(\boldsymbol{x}) \to (\rho_0(\boldsymbol{x})\rho_1(\boldsymbol{x}))^{\frac{N}{2}}$ $\mathbb{F}^{2t+1}(\boldsymbol{x}) \to (\rho_1(\boldsymbol{x})\rho_0(\boldsymbol{x}))^{\frac{N}{2}}$
$0 \leq 2d - c < 1$	$2b + c < 1$			$\mathbb{F}^t(\boldsymbol{x}) \to 0^N$
	$2b + c \geq 1$	$0 < b < \frac{1}{2} \vee 0 < c < 1$		$\mathbb{F}^t(\boldsymbol{x}) \to \lambda^N$
	$b = \frac{1}{2} \wedge c = 1$	$d = \frac{1}{2}$	N odd	$\mathbb{F}^t(\boldsymbol{x}) \to \lambda^N$
			N even	$\mathbb{F}^t(\boldsymbol{x}) \to \lambda^N$ if $\boldsymbol{x} \notin \{(01)^{\frac{N}{2}}, (10)^{\frac{N}{2}}\}$ $\mathbb{F}^t(\boldsymbol{x}) \to 0^N$ if $\boldsymbol{x} \in \{(01)^{\frac{N}{2}}, (10)^{\frac{N}{2}}\}$
		$d > \frac{1}{2}$		$\mathbb{F}^t(\boldsymbol{x}) \to \lambda^N$
	$b = \frac{1}{2} \wedge c = 0$	N odd		$\mathbb{F}^t(\boldsymbol{x}) \to 0^N$
		N even		$\mathbb{F}^{2t+\epsilon}(\boldsymbol{x}) \to (\lvert\alpha\rvert 0)^{\frac{N}{2}}$ $\mathbb{F}^{2t+1+\epsilon}(\boldsymbol{x}) \to (0\lvert\alpha\rvert)^{\frac{N}{2}}$ $, \epsilon = \begin{cases} 0 & , \alpha \geq 0 \\ 1 & , \alpha < 0 \end{cases}$
	$b = 0 \wedge c = 1$			Each segment 01^k0 has its own dynamics (see Subsection 5.6.5)

Since we only consider legal outer-totalistic ACCAs, it always holds that $\mathbb{F}^t(0^N) = 0^N$. For the configuration $x = 1^N$, the dynamics is as follows: if $2d - c = 0$ or $2d - c = 1$, then $\mathbb{F}^t(1^N) = 0^N$ or $\mathbb{F}^t(1^N) = 1^N$, respectively, while if $0 < 2d - c < 1$, then $\mathbb{F}^t(1^N)$ tends to the homogeneous configuration 0^N (if $2b + c \leq 1$) or λ^N (if $2b + c > 1$).

The results obtained show that in the set of ACCAs considered, one can observe various types of sensitivity:

- *Sensitivity to the change of a single value in an initial configuration.* This kind of sensitivity may concern all initial configurations, as for example, for $\mathbb{f}(x, y, z) = 0.4(x + z) + 0.2y$ (see (s3) in Theorem 5.4.1) or only some initial configurations, as for example, for $\mathbb{f}(x, y, z) = 0.2(x + z)y + 0.4(x + z) - 0.2y$ (see (s1) in Theorem 5.4.1).
- *Sensitivity to the change of the number of cells in the grid.* Note that if we add one cell to the considered grid, then the parity of N will change. For example, the local rule $\mathbb{f}(x, y, z) = -0.1(x + z)y + 0.5(x + z)$ is not immune to such interference in the grid structure (see Theorem 5.6.7).
- *Sensitivity to slight changes in the parameters of a local rule.* Perhaps the best example of this sensitivity are local rules parameterized by points lying on the edge P_1P_3, *i.e.*, local rules given by the following

expression:

$$\mathbb{f}(x, y, z) = -\left(\frac{1}{2} - d\right)(x + z)y + \frac{1}{2}(x + z) \ ,$$

where $0 \leq d \leq \frac{1}{2}$. The dynamics of such ACCAs for $0 \leq d < \frac{1}{2}$ (see Theorem 5.6.7) is completely different than for $d = \frac{1}{2}$ (see (s3) in Theorem 5.4.1).

Although we limited our investigation to binary initial configuration only, nearly all results can be readily extended to all configurations from $[0, 1]^N$. However, the case described in Subsection 5.6.5 would require a more substantial effort.

Solving the Relaxed One-Dimensional DCP

6

In this chapter, we show that although ACCAs cannot solve the DCP in the classical sense, it is possible to obtain a family of density-conserving ACCAs whose members are valid solutions of a relaxed version of this problem for any initial configuration, regardless of the number of cells.

6.1 Dynamical properties of density-conserving ACCAs 71
6.1.1 The case $c = 1$ 71
6.1.2 The case $0 < c < 1$ 72
6.1.3 The case $c = 0$ 76
6.2 Experimental results . . . 77
6.2.1 The case $\rho_0 = \frac{1}{2}$ 78
6.2.2 The case $\rho_0 \in (0, 1)$ 80
6.3 Conclusions 81

6.1 Dynamical properties of density-conserving ACCAs

In this section we analyze the dynamics of one-dimensional density-conserving ACCAs with unit neighborhood radius. Although our main interest lies in finding solutions of the relaxed DCP introduced in Definition 4.3.1, many of the presented facts have a more general significance. As described in Section 3.4, it is unnecessary to consider every single point of the pyramid P shown in Figure 3.2, but we can concentrate on the subset of density-conserving ACCAs whose local rules are defined by parameters $(a, b, c) \in P^I$ shown in Figure 3.4, while the discussed properties can be naturally extended to the entire pyramid P. The local rule $\mathbb{f}$ of such an ACCA is given by:

$$\mathbb{f}(x, y, z) = (a + b + c - 1)(x - z)y + (1 - a - c)x + cy + az \ , \qquad (6.1)$$

where $(a, b, c) \in P^I$. Hence, for any $\rho \in [0, 1]$, the point $(\rho, \rho, \ldots, \rho)$ is a fixed point of $\mathbb{F}$. Note that in the pyramid P (and also P^I) the parameter c varies from 0 (base of the pyramid) to 1 (top of the pyramid).

6.1.1 The case $c = 1$

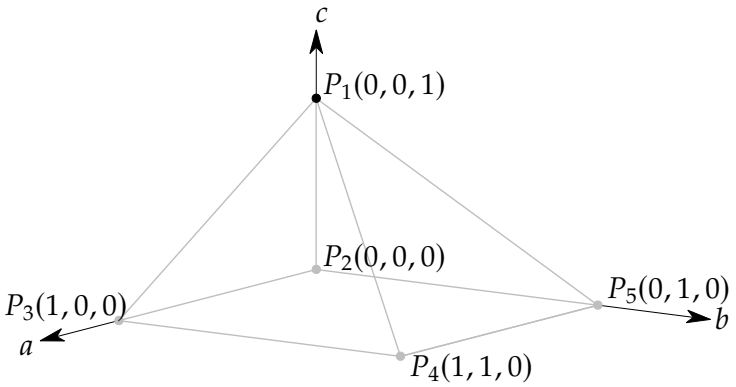


Figure 6.1: Parameterization of the set of one-dimensional density-conserving ACCAs with unit neighborhood radius with selected ACCA for the case $c = 1$.

In this case, there is only one density-conserving rule, *i.e.*, the identity rule (ECA204), found at the top vertex of the pyramid P as presented in Figure 6.1. This rule is not able to solve the DCP, irrespective of the formulation (Defs. 4.2.1 and 4.3.1), since every configuration $\boldsymbol{x}$ is a fixed point of $\mathbb{F}$. Yet, as we will show in the following parts, for c very close to one, the behavior of the corresponding ACCAs is strictly different

from that of the identity rule, which is very interesting and might be somewhat surprising.

6.1.2 The case $0 < c < 1$

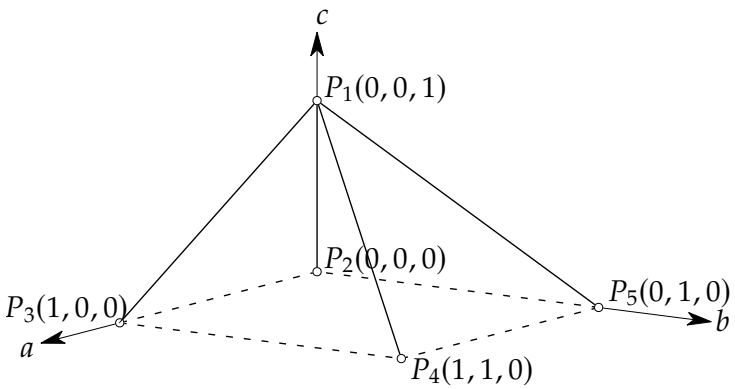


Figure 6.2: Parameterization of the set of one-dimensional density-conserving ACCAs with unit neighborhood radius with selected ACCAs for the case $0 < c < 1$.

Local rules in this case can be found anywhere in P, except for its bottom plane and apex as presented in Figure 6.2. The following theorem implies that for any $\rho \in [0,1]$ the point $(\rho, \rho, \ldots, \rho)$ is an attractor of $\mathbb{F}$ and characterizes its basin.

Theorem 6.1.1 *Let $\mathbb{f}$ be the local rule of a one-dimensional density-conserving ACCA parameterized by $(a,b,c) \in P$ and let $\mathbb{F}$ be the corresponding global rule. If $0 < c < 1$, then for any $\boldsymbol{x} \in [0,1]^*$, it holds that:*

$$\lim_{t\to\infty} \mathbb{F}^t(\boldsymbol{x}) = (\rho(\boldsymbol{x}), \rho(\boldsymbol{x}), \ldots, \rho(\boldsymbol{x})) \,. \tag{6.2}$$

The proof of Theorem 6.1.1 follows from four lemmas that are presented and proven below. As mentioned before, we will restrict the proof of Theorem 6.1.1 to P^I, but extending it from P^I to P is straightforward by using the transformations defined in Eqs. (2.4) and (2.5).

The general idea for the proof is based on the previous works on Fuzzy CAs presented in [48].

Lemma 6.1.2 *Let $\mathbb{f}$ be the local rule of a one-dimensional density-conserving ACCA parameterized by $(a,b,c) \in P^I$, then $\mathbb{f}$ is increasing in the first and in the third variable.*

Proof. The following lower bounds on the partial derivatives of $\mathbb{f}$ hold as $a + b + c \leq 1$:

$$\begin{aligned} \mathbb{f}'_x(x,y,z) &= (a+b+c-1)y + (1-a-c) \\ &= (1-a-c)(1-y) + by \geq b(1-y) + by = b \geq 0 \,, \end{aligned}$$

and

$$\begin{aligned} \mathbb{f}'_z(x,y,z) &= -(a+b+c-1)y + a \\ &= (1-b-c)y + a(1-y) \geq ay + a(1-y) = a \geq 0 \,, \end{aligned}$$

which suffices to prove our assertion. □

Lemma 6.1.3 *Let $\mathbb{f}$ be the local rule of a one-dimensional density-conserving ACCA parameterized by $(a,b,c) \in P^I$. If $(x,y,z) \in [m,M]^3$, for some $0 \leq m \leq M \leq 1$, then $f(x,y,z) \in [m,M]$.*

Proof. From Lemma 6.1.2, it follows that:

$$\mathbb{f}(x,y,z) \geq \mathbb{f}(m,y,m) = (1-c)m + cy \geq (1-c)m + cm = m \ ,$$

and

$$\mathbb{f}(x,y,z) \leq \mathbb{f}(M,y,M) = (1-c)M + cy \leq (1-c)M + cM = M \ ,$$

which completes the proof. □

Lemma 6.1.4 *Let $\mathbb{f}$ be the local rule of a one-dimensional density-conserving ACCA parameterized by $(a,b,c) \in P^I$. If $0 \leq m \leq m' \leq M \leq 1$, then the following conditions hold:*

$$\begin{aligned}
&\mathbb{f}(m,m,M) \geq m + (M-m)(a + (1-a-b-c)m) \ , \\
&\mathbb{f}(m,M,M) \geq m + (M-m)c \ , \\
&\mathbb{f}(M,M,m) \geq m + (M-m)c \ , \\
&\mathbb{f}(m,M,m') \geq m + (m'-m)c \ , \\
&\mathbb{f}(m',M,m) \geq m + (m'-m)c \ .
\end{aligned}$$

Proof. Relying on the fact that $a + b + c \leq 1$, we easily find that:

$$\begin{aligned}
\mathbb{f}(m,m,M) &= (a+b+c-1)(m-M)m + (1-a-c)m + cm + aM \\
&= (1-a-b-c)(M-m)m + m + (M-m)a \\
&\geq m + (M-m)(a + (1-a-b-c)m) \ , \\
\mathbb{f}(m,M,M) &= (a+b+c-1)(m-M)M + (1-a-c)m + cM + aM \\
&\geq (1-a-c)m + cM + aM \geq m + (M-m)c \ , \\
\mathbb{f}(M,M,m) &= (a+b+c-1)(M-m)M + (1-a-c)M + cM + am \\
&\geq (a+b+c-1)(M-m) + (1-a)M + am \\
&= m + (M-m)(b+c) \geq m + (M-m)c \ , \\
\mathbb{f}(m,M,m') &= (a+b+c-1)(m-m')M + (1-a-c)m + cM + am \\
&\geq (1-a-c)m + cM + am' \geq m + (m'-m)c \ , \\
\mathbb{f}(m',M,m) &= (a+b+c-1)(m'-m)M + (1-a-c)m' + cM + am \\
&\geq (a+b+c-1)(m'-m) + (1-a-c)m' + cm' + am \\
&= m + (m'-m)(b+c) \geq m + (m'-m)c \ .
\end{aligned}$$

□

Lemma 6.1.5 *Let $\mathbb{f}$ be the local rule of a one-dimensional density-conserving ACCA parameterized by $(a,b,c) \in P^I$. Suppose that a configuration $\boldsymbol{x}$ belongs to $[m,M]^n$ for some $0 \leq m \leq M \leq 1$ and $n \in \mathbb{N}$. If there exists an $i_0 \in \{0,\ldots,n-1\}$ such that $x_{i_0} = M$, then for any $i \in \{0,1,\ldots,n-1\}$,*

it holds that:

$$x_i^{n-1} \geq m + (M-m)\gamma^{n-1} ,$$

where $\gamma = \min(c, a + m(1-a-b-c))$.

Proof. Let us recall that $\mathbb{f}$ is affine in each variable. Additionally, we use the fact that $\mathbb{f}$ is increasing in the first and third variable (Lemma 6.1.2) and the lower bounds obtained in Lemma 6.1.4.

First, let us note that:

$$x_{i_0}^1 = \mathbb{f}(x_{i_0-1}, M, x_{i_0+1}) \geq \mathbb{f}(m, M, m) = m + (M-m)c ,$$

and

$$\begin{aligned} x_{i_0-1}^1 &= \mathbb{f}(x_{i_0-2}, x_{i_0-1}, M) \\ &\geq \min\left(\mathbb{f}(m, m, M), \mathbb{f}(m, M, M)\right) \\ &\geq \min\left(m + (M-m)(a + m(1-a-b-c))m + (M-m)c\right) , \end{aligned}$$

which means that both $x_{i_0}^1$ and $x_{i_0-1}^1$ are not smaller than $m + (M-m)\gamma$. We now refine this result and prove that if for some $t \geq 0$ both x_i^t and x_{i+1}^t are not smaller than $m + (M-m)\gamma^k$, then $x_{i-1}^{t+1}, x_i^{t+1}, x_{i+1}^{t+1}$ are not smaller than $m + (M-m)\gamma^{k+1}$. Indeed, if we denote $m' = m + (M-m)\gamma^k$, we have:

$$\begin{aligned} x_{i-1}^{t+1} &= \mathbb{f}(x_{i-2}^t, x_{i-1}^t, x_i^t) \\ &\geq \min\left(\mathbb{f}(m, m, m'), \mathbb{f}(m, M, m')\right) \geq m + (m'-m)\gamma , \\ x_i^{t+1} &= \mathbb{f}(x_{i-1}^t, x_i^t, x_{i+1}^t) \\ &\geq \min\left(\mathbb{f}(m, m', m'), \mathbb{f}(m, M, m')\right) \geq m + (m'-m)\gamma , \\ x_{i+1}^{t+1} &= \mathbb{f}(x_i^t, x_{i+1}^t, x_{i+2}^t) \\ &\geq \min\left(\mathbb{f}(m', m', m), \mathbb{f}(m', M, m)\right) \geq m + (m'-m)\gamma , \end{aligned}$$

and, of course, $m + (m'-m)\gamma = m + (M-m)\gamma^{k+1}$. Using induction finitely many times, the lemma is proven. □

Having proven Lemmas 6.1.3–6.1.5, we can complete the proof of Theorem 6.1.1.

Proof of Theorem 6.1.1. Let us choose $n \in \mathbb{N}$ and $\boldsymbol{x} \in [0,1]^n$. We define two sequences, namely $(m_t)_{t\geq 0}$ and $(M_t)_{t\geq 0}$, where

$$m_t = \min(x_0^t, x_1^t, \ldots, x_{n-1}^t)$$

and

$$M_t = \max(x_0^t, x_1^t, \ldots, x_{n-1}^t) .$$

According to Lemma 6.1.3, we know that the sequence (m_t) is increasing, while the sequence (M_t) is decreasing, *i.e.*:

$$m_0 \leq m_1 \leq m_2 \leq \ldots \leq M_2 \leq M_1 \leq M_0 .$$

Hence, both sequences converge. Let $m = \lim_{t\to\infty} m_t$ and $M = \lim_{t\to\infty} M_t$. If we can prove that $m = M$, then the assertion follows due to the fact that $\mathbb{f}$ conserves density. To obtain a contradiction, suppose that $M - m = \varepsilon > 0$.

Let us start with the case $a > 0$, for which it holds that:

$$\gamma = \min\left(c, a + (1 - a - b - c)m_0\right) > 0 \ .$$

Hence, there exists a $t_0 \geq 0$ such that:

$$M_{t_0} - m_{t_0} < \frac{\varepsilon}{1 - \gamma^{n-1}} \ .$$

For some $i_0 \in \{0, 1, \ldots, n-1\}$, we have $x_{i_0}^{t_0} = M_{t_0}$. Thus, from Lemma 6.1.5 and the fact that $m_0 \leq m_{t_0}$, we conclude that for each $i \in \{0, 1, \ldots, n-1\}$, we have:

$$x_i^{t_0+n-1} \geq m_{t_0} + (M_{t_0} - m_{t_0})\gamma^{n-1} \ ,$$

but this means that:

$$m_{t_0+n-1} \geq m_{t_0} + (M_{t_0} - m_{t_0})\gamma^{n-1} \ .$$

Consequently:

$$\begin{aligned}
\varepsilon = M - m &\leq M_{t_0} - m_{t_0+n-1} \\
&\leq M_{t_0} - \left(m_{t_0} + (M_{t_0} - m_{t_0})\gamma^{n-1}\right) \\
&= \left(1 - \gamma^{n-1}\right)\left(M_{t_0} - m_{t_0}\right) \\
&< \left(1 - \gamma^{n-1}\right)\frac{\varepsilon}{1 - \gamma^{n-1}} = \varepsilon \ ,
\end{aligned}$$

which is impossible. This concludes the proof in the case $a > 0$. The case $a = 0$ is more complicated, because we do not know whether $\gamma > 0$ or not. However, in this case $\gamma = \min(c, (1 - c)m_0)$ because if $(a, b, c) \in P^I$ then $0 \leq b \leq a$. As this depends on m_0, taking into account the above arguments, we know that even for $a = 0$ each configuration $\boldsymbol{x}$ with $m_0 > 0$ evolves to a homogeneous one. Hence, the proof will be concluded if we show that for each configuration $\boldsymbol{x}$ belonging to $[0, 1]^n \setminus \{0\}^n$, there exists $t \in \mathbb{N}$ such that all x_i^t are greater than 0.

Let us note that if $a = 0$ and $b = 0$, then the LUT of $\mathbb{f}$ is relatively simple (Table 6.1). It is easily seen that the local rule $\mathbb{f}$ fills a cell i with 0 only

$(0,0,0)$	$(0,0,1)$	$(0,1,0)$	$(0,1,1)$	$(1,0,0)$	$(1,0,1)$	$(1,1,0)$	$(1,1,1)$
l_0	l_1	l_2	l_3	l_4	l_5	l_6	l_7
0	0	c	1	$1-c$	$1-c$	c	1

Table 6.1: The LUT of a one-dimensional density-conserving ACCA parameterized by $(0, 0, c)$.

when the state of cells i and $i - 1$ were 0 at the previous time step. This means that if the initial configuration contains at least one substring of 1s, then in each subsequent step the length of each substring of 0s decreases by one. Consequently, after at most $n - 1$ steps, all 0s will disappear and m_{n-1} will be greater than 0, thereby concluding the proof for the case $a = 0$. □

From the convergence given by Eq. (6.2), it follows that, for any $\boldsymbol{x} \in [0, 1]^n$ and any $\epsilon > 0$, there exists a T such that:

$$\mathbb{F}^T(\boldsymbol{x}) \in [\rho(\boldsymbol{x}) - \epsilon, \rho(\boldsymbol{x}) + \epsilon]^n \ .$$

Let $\rho_0 \in (0, 1)$. We see that if $\rho(\boldsymbol{x}) < \rho_0$, then for any $0 < \epsilon < \rho_0 - \rho(\boldsymbol{x})$, it holds that:

$$[\rho(\boldsymbol{x}) - \epsilon, \rho(\boldsymbol{x}) + \epsilon]^n \subset [0, \rho_0)^n \ .$$

Analogously, if $\rho(\boldsymbol{x}) > \rho_0$, then for any $0 < \epsilon < \rho(\boldsymbol{x}) - \rho_0$, it holds that:

$$[\rho(\boldsymbol{x}) - \epsilon, \rho(\boldsymbol{x}) + \epsilon]^n \subset (\rho_0, 1]^n \ .$$

In other words, the condition necessary for solving the relaxed DCP is fulfilled. Moreover, we see that the condition fulfilled is even stronger than the one in Definition 4.3.1, since it is satisfied for every initial configuration in $[0,1]^*$. As a consequence we get the following fact.

Proposition 6.1.6 *Every one-dimensional density-conserving ACCA whose local rule is parameterized by $(a,b,c) \in P$ with $c \in (0,1)$, is a solution of the relaxed DCP at any threshold $\rho_0 \in (0,1)$.*

6.1.3 The case $c = 0$

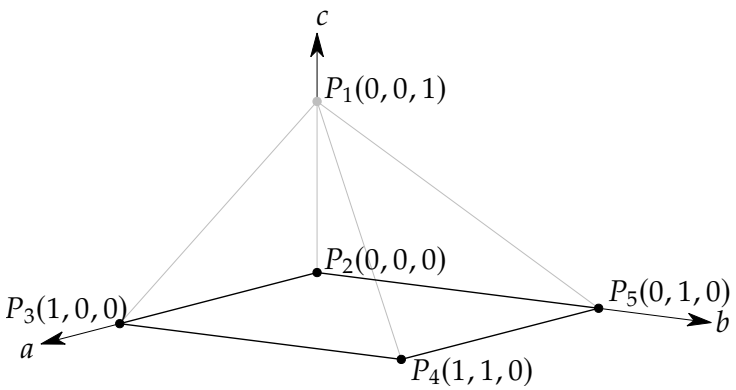


Figure 6.3: Parameterization of the set of density-conserving ACCAs with unit neighborhood radius with selected ACCAs for the case $c = 0$.

This case concerns the rules that can be found in the bottom plane of P as presented in Figure 6.3. Any local rule $\mathbb{f}$ of a density-conserving ACCA can be expressed as:

$$\mathbb{f}(x,y,z) = (a+b-1)(x-z)y + (1-a)x + az \ ,$$

which yields, in particular, $\mathbb{f}(1,y,1) = 1$. Hence, in general, $\mathbb{f}$ cannot solve the DCP in the relaxed sense. Indeed, let n be an even number ($n > 4$) and choose an initial configuration $\boldsymbol{x} \in \{0,1\}^n$ as follows:

$$x_i = \begin{cases} 1 & \text{if } i \text{ is even} \ , \\ 1 & \text{if } i = 1 \ , \\ 0 & \text{otherwise} \ , \end{cases}$$

i.e., $\boldsymbol{x} = 1110101010\ldots10$, then $x_i^t = 1$ if $i \equiv t \pmod 2$. If for some t the value of each cell would be greater than $\frac{1}{2}$, then:

$$\rho(\boldsymbol{x}^t) \geq \frac{1 \cdot \frac{1}{2}n + \frac{1}{2} \cdot \frac{1}{2}n}{n} = \frac{1}{2} + \frac{1}{4} > \frac{1}{2} + \frac{1}{n} = \rho(\boldsymbol{x}) \ ,$$

which contradicts the fact that the ACCA defined by $\mathbb{f}$ is density-conserving. Hence, none of the ACCAs parameterized by $(a,b,0) \in P$ can solve the DCP in the relaxed sense for this particular initial configuration. Following the approach used in the case $0 < c < 1$, it can be shown that if $0 < a, b < 1$, then for any odd number n and any configuration

$\boldsymbol{x} \in \{0,1\}^n$, it holds that:

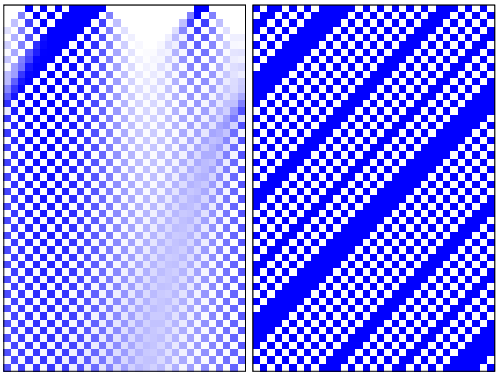
$$\lim_{t\to\infty} \mathbb{F}^t(\boldsymbol{x}) = (\rho(\boldsymbol{x}))^n \ . \tag{6.3}$$

Furthermore, if $a \in \{0,1\}$ or $b \in \{0,1\}$, then even for odd n it is possible to find a configuration $\boldsymbol{x} \in \{0,1\}^n$ for which the condition in Eq. (6.3) is not satisfied, and hence, for which we cannot solve the relaxed DCP. To show this, let us consider $b = 0$, *i.e.*, the case of the edge P_2P_3 in pyramid P. Then, for any $a \in [0,1]$, the local rule $\mathbb{f}$ corresponding to the point $(a,0,0) \in P$ is given by the LUT shown in Table 6.2.

$(0,0,0)$	$(0,0,1)$	$(0,1,0)$	$(0,1,1)$	$(1,0,0)$	$(1,0,1)$	$(1,1,0)$	$(1,1,1)$
l_0	l_1	l_2	l_3	l_4	l_5	l_6	l_7
0	a	0	1	$1-a$	1	0	1

Table 6.2: The LUT of a one-dimensional density-conserving ACCA parameterized by $(a,0,0)$.

Note that such a rule behaves as the left-shift ECA rule for all configurations whose zeros are isolated, *i.e.*, configurations in which there are no substrings of the form $(0)^j$ for $j > 1$. The behavior for an example rule parameterized by $(\frac{1}{2},0,0)$ is presented in Figure 6.4.



(a) **(b)**

Figure 6.4: Space-time diagrams of a one-dimensional density-conserving rule parameterized by $(\frac{1}{2},0,0)$ and a configuration with (a) and without (b) substrings of the form $(0)^j$, $j > 1$.

For such configurations, the LUT entries l_1 and l_4 are never used when evolving the ACCA and since the remaining entries agree with the shift-left ECA, it has the same behavior as this ECA, which is clearly not a solution of the relaxed DCP. A similar analysis applies to the remaining cases, *i.e.*, the edges P_3P_4, P_4P_5, P_2P_5, so that we may conclude that there are no solutions of the relaxed DCP for $c = 0$.

6.2 Experimental results

In this section, we analyze how good one-dimensional density-conserving ACCAs with unit neighborhood radius are at solving the relaxed DCP. For a given n, we measure the convergence time T (*i.e.*, the minimal value of T rendering Eq. (4.1) true for the given n). Since the convergence time can be very high, for practical reasons we halt the simulation after Λ iterations. As presented in Section 3.4, the one-dimensional density-conserving ACCAs are parameterized by $(a,b,c) \in P$ (Figure 3.2), defining the LUT in Eq. (3.14). We sample the pyramid P with points (a_i,b_j,c_k), $i,j \leq k$, $i,j,k \in \{0,\ldots,100\}$, where the coordinates, a_p, b_p, c_p, are given by $0.01\,p$ and $a_i, b_j \leq 1 - c_k$.

Two experiments are carried out. The goal of the first experiment is to determine the convergence time of the investigated ACCA family in the case of the most commonly used threshold $\rho_0 = \frac{1}{2}$. Since only one threshold is considered, we are able to explore two relatively large sets of initial configurations, as such, we can analyze the impact of the configuration length on the convergence time. In the second experiment we compare the convergence time for different thresholds in the set $\{0.001, 0.002, \ldots, 0.999\}$.

6.2.1 The case $\rho_0 = \frac{1}{2}$

The choice of the threshold $\frac{1}{2}$ is motivated by the fact that the classical DCP is typically considered with this threshold. The experiment is conducted for two different sets of initial configurations, referred to as $\mathcal{I}_{25}$ and $\mathcal{I}_{149}$. The former set $\mathcal{I}_{25}$ contains all binary configurations of $n = 25$ cells. In this case, we put $\Lambda = 1000$. With these settings and in agreement with the theoretical findings in Section 6.1, the simulations confirm that all ACCAs corresponding to the points in the interior of the pyramid P solve the relaxed DCP correctly in less than Λ iterations. The best ACCA needs at most 73 iterations to solve the relaxed DCP for any configuration from this set of initial configurations. The local rule $\mathbb{f}$ of this solution is parameterized by the point $(0, 0, 0.05)$ in the pyramid P and is very close to the traffic-right rule. Due to the symmetries defined in Eqs. (2.4) and (2.5), the same convergence time is obtained for local rules parameterized by $(0.01, 0.01, 0.05)$, $(0.94, 0.94, 0.05)$ and $(0.95, 0.95, 0.05)$, corresponding to $\mathbb{f}^I$, $\mathbb{f}^C$ and $\mathbb{f}^{IC}$, respectively. The convergence times for ACCAs parametrized by points in the horizontal cross-section of the pyramid P containing the best solution are presented in Figure 6.5.

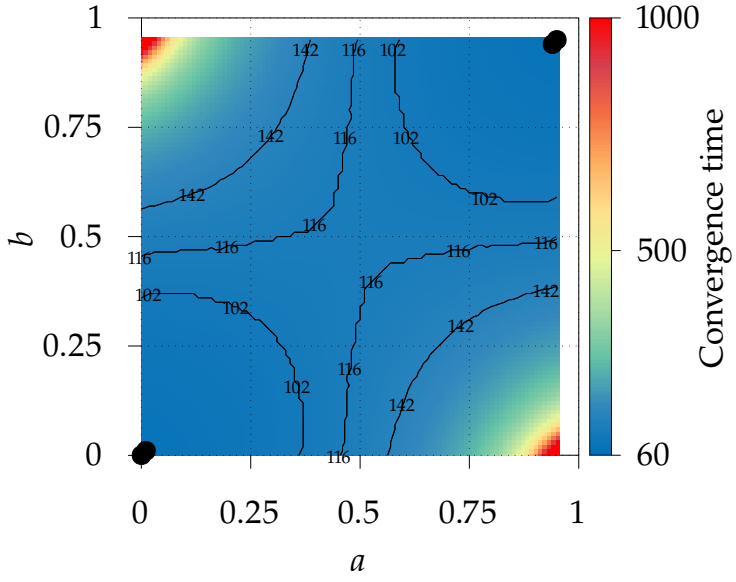


Figure 6.5: Convergence times obtained for density-conserving ACCAs parameterized by points in the horizontal cross-section of the pyramid P that contains the best solution (black dot) for $n = 25$ cells and cross-section at $c = 0.05$.

Furthermore, Figure 6.6 shows that the ACCAs that require at most 77 iterations to solve the relaxed DCP are positioned near the best ACCA.

The set $\mathcal{I}_{149}$ contains $2 \cdot 10^6$ initial configurations that have been randomly selected from the set of all binary configurations with $n = 149$ cells. The time limit is set to $\Lambda = 20000$. Similarly to the case of $n = 25$

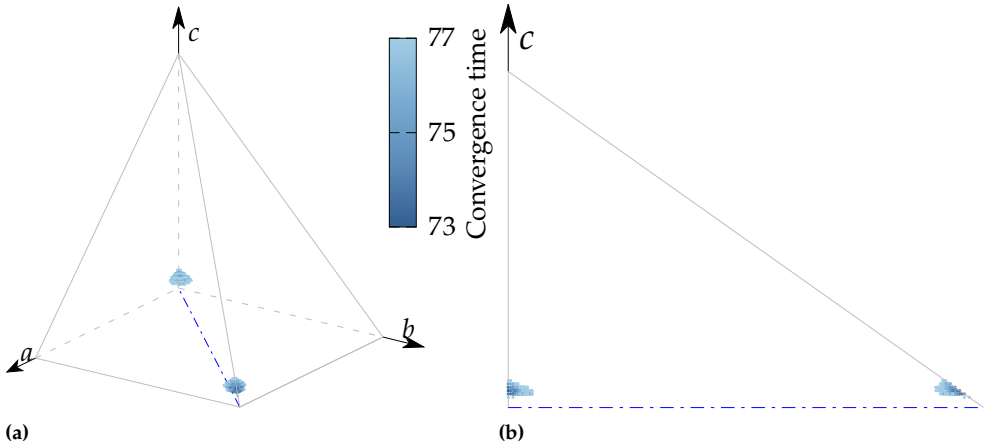


Figure 6.6: Location of the points in the pyramid P corresponding to ACCAs with low convergence time for threshold $\rho_0 = 0.5$ and for configurations of $n = 25$ cells: (a) in the entire pyramid P, and (b) for the diagonal cross-section.

cells, the simulations show that every ACCA defined by a point in the interior of the pyramid P is a solution of the relaxed DCP. The best solution found in this case is able to solve the relaxed DCP in at most 2918 iterations (maximum over the set of initial configurations), and its local rule is parameterized by the point $(0.0, 0.0, 0.02)$. Due to symmetry, the same convergence time is obtained for the rule parameterized by $(0.98, 0.98, 0.02)$. The convergence times for ACCAs parametrized by points in the cross-section of pyramid P containing the best solution ($c = 0.02$) are presented in Figure 6.7.

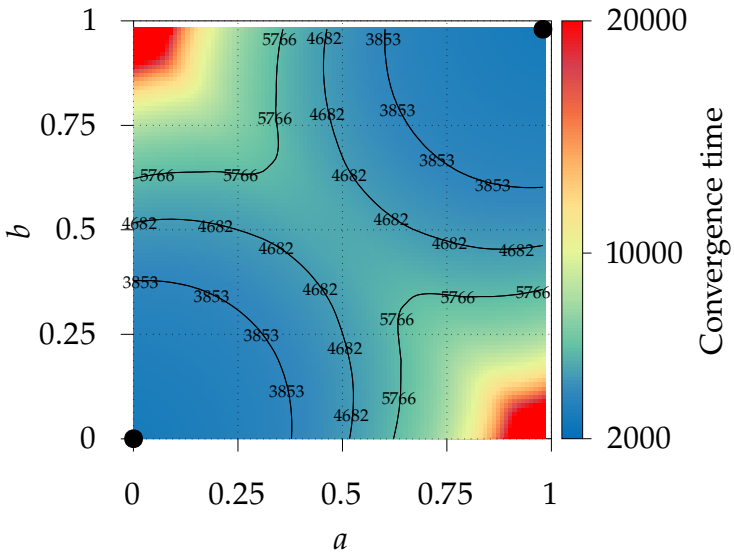


Figure 6.7: Convergence times obtained for density-conserving ACCAs parameterized by points in the horizontal cross-section of the pyramid P that contains the best solution (black dot) for $n = 149$ cells and cross-section at $c = 0.02$.

Summarizing, the results of this experiment show that, although most of the density-conserving ACCAs solve the relaxed DCP at threshold $\rho_0 = \frac{1}{2}$, the convergence time increases and the location of the ACCAs with the lowest convergence times in the pyramid P changes as the length of the initial configuration increases.

6.2.2 The case $\rho_0 \in (0, 1)$

In this experiment we consider the set of all initial configurations with $n = 23$ cells ($\mathcal{I}_{23}$). As for the previous experiment, we put $\Lambda = 1000$. For every ACCA parametrized by a point in the pyramid P, we measure the convergence time for a series of relaxed DCPs with thresholds from the set $\{0.001, \ldots, 0.999\}$. The simulations are organized in such a way that each of the ACCAs has to evolve only once for every initial configuration, by evolving every ACCA for Λ consecutive time steps and capturing the statistics for all considered thresholds.

A summary of the results is presented in Figure 6.8, where the color scheme refers to the threshold. Figure 6.8a shows the location of the

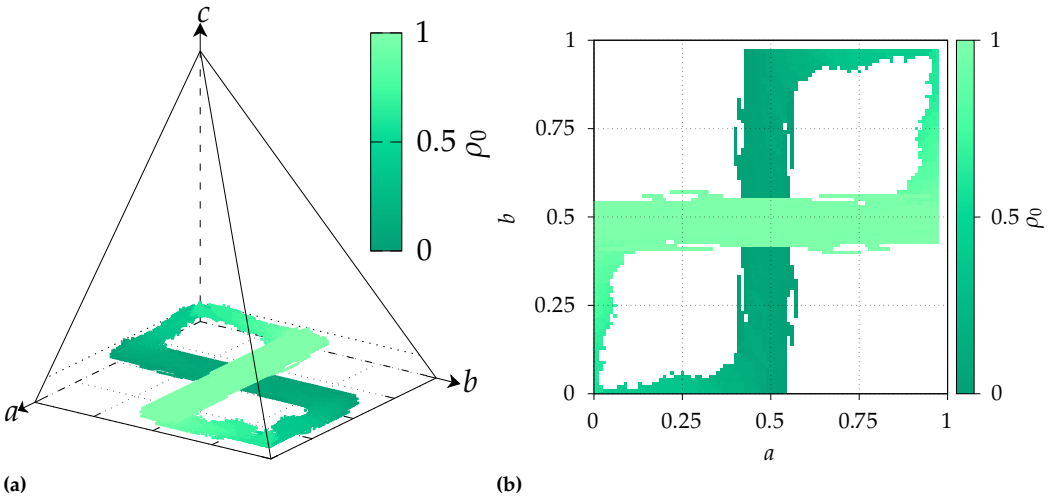


Figure 6.8: Location of the points in the pyramid P corresponding to ACCAs with the lowest convergence time for a given threshold ρ_0, for configurations with $n = 23$ cells: (a) in the entire pyramid P, and (b) for the cross-section at $c = 0.03$.

points in the pyramid P corresponding to the ACCAs with the lowest convergence time. It shows that, in general, the best solutions are located in the bottom part of the pyramid P, irrespective of the threshold. Moreover, it shows that for thresholds close to 0 and 1, there is a relatively high number of highly performant solutions, while for thresholds close to $\frac{1}{2}$, this number is low. Also, there is an area in the middle of the cross-section where there exist good solutions for low and high thresholds. A detailed view of one of the cross-sections ($c = 0.03$) of the pyramid P is shown in Figure 6.8b. The results of the previous experiment show that the convergence time depends on the length of the initial configuration, while the results of this experiment show that also the threshold influences the performance.

For a given threshold ρ_0, we calculated the minimum convergence time over all density-conserving ACCAs (Figure 6.9, marked in black). This relation is rather complex, but this can be explained by the fact that we only considered binary initial configurations with $n = 23$ cells. Thus, the initial density is always equal to $\frac{k}{23}$ for some $k = 0, 1, \ldots, 23$. The peaks of the curve correspond to thresholds that are very close to $\frac{k}{23}$. So, this means that the difficulty of solving the relaxed DCP grows when starting from an initial configuration with a density close to the threshold. This finding corresponds to earlier ones on binary CAs in the classical DCP,

where the most challenging initial configurations are those in which the number of 0s and 1s are almost equal [74].

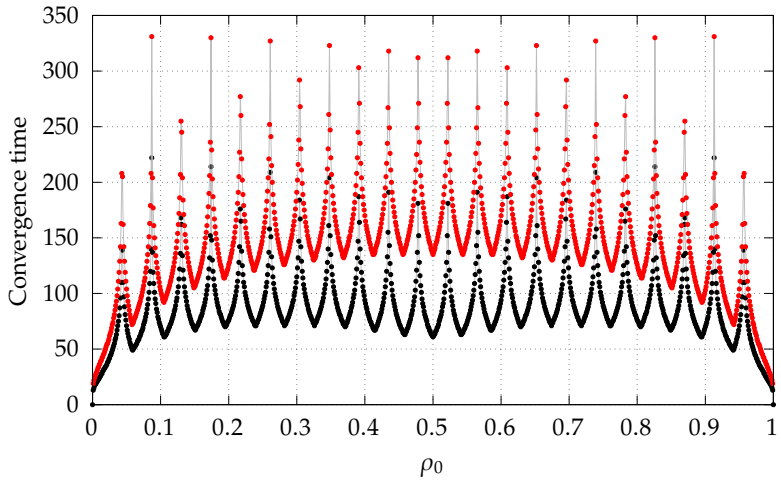


Figure 6.9: Minimum convergence time for a given threshold ρ_0 over all ACCAs (black) compared with the convergence time of the diffusion ACCA (red).

Furthermore, for each threshold ρ_0, we calculated the convergence time for the diffusion ACCA with local rule $\mathbb{f}_d$ defined by Eq. (2.7). The result is presented in Figure 6.9 (red). For each threshold, the rule $\mathbb{f}_d$ turns out to be much worse than the best found ACCA. In particular, for $\rho_0 = \frac{1}{2}$ the convergence time is about the double of that of the best ACCA found.

Summarizing, the outcome of this experiment is that the convergence time strongly depends on the threshold and that this relation is not linear and not trivial. Moreover, the best ACCAs are parameterized by points of the pyramid P that are close to its base, irrespective of the threshold.

6.3 Conclusions

The results presented in this chapter justify that it is worthwhile to explore CCA-based models for hard computation tasks like the DCP. In contrast to classical CAs, there exists a class of density-conserving ACCAs solving the DCP in the relaxed sense, confirming that the computational capabilities of CCAs supersede those of classical CAs. Our experimental results highlight that the convergence time of such classifiers depends on the number of cells in the initial configuration and the selected threshold. In the latter case, the relation is clearly not linear.

Solving the Fixed-Length One-Dimensional DCP

7

In this chapter, we consider the classical DCP in the context of ACCAs. As we proved in Chapter 4, there exists no general solution to this problem that is valid for any number of cells. Therefore, we consider this problem in the case of a fixed number of cells. We start with finding necessary conditions for an ACCA to be a candidate of a classifier. Then, based on this theoretical knowledge, we evaluate experiments for 23 cells.

7.1 Problem formulation 83
7.2 Necessary conditions 86
7.3 Sufficient stopping conditions 91
7.4 Simulation results 93
7.5 Conclusions 96

7.1 Problem formulation

For the local rule $\mathbb{f}$ of an ACCA, we introduce three parameters η, ϵ and γ:

$$\begin{aligned} \eta &= 1 - (l_1 + l_2 + l_4) \,, \\ \epsilon &= l_3 + l_5 + l_6 - 2 \,, \\ \gamma &= l_2 + l_5 - 1 \,. \end{aligned} \tag{7.1}$$

Note that $l_i \in [0,1]$ yields $\gamma \in [-1,1]$ and $\eta, \epsilon, \eta+\gamma, \epsilon-\gamma \in [-2,1]$. Moreover, as Theorem 3.4.1 states, an ACCA conserves density if and only if its local rule is double-quiescent and satisfies $\eta = \epsilon = \gamma = 0$.

If η, ϵ, γ are fixed, then Eq. (7.1) allows to parametrize the set of all double-quiescent local rules of ACCAs with these parameters in the following way:

$$\begin{cases} l_0 = 0 \\ l_1 = a \\ l_2 = c \\ l_3 = 1 - b \\ l_4 = 1 - \eta - a - c \\ l_5 = 1 + \gamma - c \\ l_6 = \epsilon - \gamma + b + c \\ l_7 = 1 \end{cases} \text{, where } \begin{cases} a, b, c \in [0,1] \\ \gamma \le c \le 1 + \gamma \\ -\eta \le a + c \le 1 - \eta \\ \gamma - \epsilon \le b + c \le 1 + \gamma - \epsilon \end{cases} \tag{7.2}$$

We can represent the solution set graphically in the Cartesian coordinate system. For example, the set of all double-quiescent local rules with $\eta = 0.1$, $\epsilon = 0.2$, $\gamma = 0.3$ are the local rules with a LUT of the form

$$(0,\ a,\ c,\ 1-b,\ 0.9-a-c,\ 1.3-c,\ -0.1+b+c,\ 1) \,,$$

where the point (a, b, c) belongs to the polyhedron shown in Figure 7.1.

To prove some of the technical lemmas, we introduce the following notation: for $\boldsymbol{x} \in \{0,1\}^N$, $n_i(\boldsymbol{x})$ denotes the number of occurrences of the neighborhood N_i in $\boldsymbol{x}$, for $i \in \{0, 1, \ldots, 7\}$. In the remainder, $\boldsymbol{x}$ will be omitted if there is no ambiguity. Clearly, for each $\boldsymbol{x} \in \{0,1\}^N$, it holds

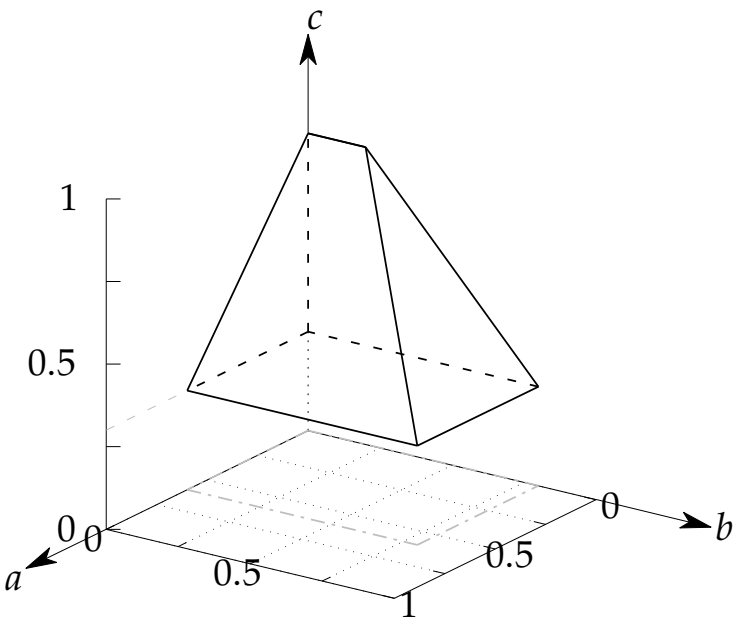


Figure 7.1: Parameterization of the set of double-quiescent local rules with parameters $\eta = 0.1$, $\epsilon = 0.2$, $\gamma = 0.3$.

that $n_0 + n_1 + \ldots + n_7 = N$, but there are other dependencies among the n_i (see for instance [45]):

$$\begin{aligned} n_1 &= n_4 \,, \\ n_3 &= n_6 \,, \\ n_1 + n_5 &= n_2 + n_3 \,. \end{aligned}$$

From this, we obtain a very useful way to calculate the density of a binary configuration and the density of its image under $\mathbb{F}$.

Lemma 7.1.1 *Let $\mathbb{f}$ be the local rule of a one-dimensional ACCA with unit radius and let $\mathbb{F}$ be the corresponding global rule. If $\mathbb{f}$ is double-quiescent, then for any $\boldsymbol{x} \in \{0,1\}^N$, it holds that*

$$\rho(\boldsymbol{x}) = \frac{1}{N}\left(n_2 + 2n_3 + n_7\right) \,,$$

and, moreover,

$$\rho(\mathbb{F}(\boldsymbol{x})) = \frac{1}{N}\Big[(n_2 + 2n_3 + n_7) - \eta \cdot n_1 + \gamma \cdot (n_2 - n_1) + \epsilon \cdot n_3\Big] \,.$$

Proof. As $n_4 = n_1$, $n_6 = n_3$ and $n_5 = n_2 + n_3 - n_1$, we have

$$\begin{aligned} \rho(\boldsymbol{x}) &= \frac{1}{3N}\Big(0{\cdot}n_0 + 1{\cdot}n_1 + 1{\cdot}n_2 + 2{\cdot}n_3 + 1{\cdot}n_4 + 2{\cdot}n_5 + 2{\cdot}n_6 + 3{\cdot}n_7\Big) \\ &= \frac{1}{3N}\Big(n_1 + n_2 + 2n_3 + n_1 + 2(n_2 + n_3 - n_1) + 2n_3 + 3n_7\Big) \\ &= \frac{1}{N}\Big(n_2 + 2n_3 + n_7\Big) \,, \end{aligned}$$

which shows the first part of lemma. Moreover,

$$\rho(\mathbb{F}(\boldsymbol{x})) = \frac{1}{N}\Big(l_0 n_0 + l_1 n_1 + l_2 n_2 + l_3 n_3 + l_4 n_4 + l_5 n_5 + l_6 n_6 + l_7 n_7\Big)$$

$$= \frac{1}{N}\Big(l_1 n_1 + l_2 n_2 + l_3 n_3 + l_4 n_1 + l_5(n_2 + n_3 - n_1) + l_6 n_3 + n_7\Big)$$
$$= \frac{1}{N}\Big[(n_2 + 2n_3 + n_7) - \eta \cdot n_1 + \gamma \cdot (n_2 - n_1) + \epsilon \cdot n_3\Big] ,$$

which shows the second part. □

This lemma shows that for a given $\boldsymbol{x} \in \{0,1\}^N$, the difference $\rho(\mathbb{F}(\boldsymbol{x})) - \rho(\boldsymbol{x})$ depends only on η, ϵ and γ and does not depend on the LUT entries l_i separately, which motivates why we study these parameters in the context of the dynamics of a given ACCA.

Next, we recall the fundamental definitions for the discussed topic, which we formulated in Section 4.2, but now we consider in particular $\rho_0 = \frac{1}{2}$.

Definition 7.1.1 *Let $\mathbb{f}$ be the local rule of a CCA, $\mathbb{F}$ be the corresponding global rule and N be a given natural number. We say that this CCA solves the DCP for N and we write $\mathbb{f} \in \mathrm{DCP}(N)$, if for any $\boldsymbol{x} \in \{0,1\}^N$ the following implications hold:*

$$\begin{aligned} \rho(\boldsymbol{x}) < \frac{1}{2} &\Rightarrow \lim_{t\to\infty} \mathbb{F}^t(\boldsymbol{x}) = 0^N , \\ \rho(\boldsymbol{x}) > \frac{1}{2} &\Rightarrow \lim_{t\to\infty} \mathbb{F}^t(\boldsymbol{x}) = 1^N . \end{aligned} \tag{7.3}$$

Note that a CCA solves the DCP in the classical sense if its local rule satisfies $\mathbb{f} \in \bigcap_{N=1}^{\infty} \mathrm{DCP}(N)$.

In Definition 7.1.1, the constant $\frac{1}{2}$ acts as a threshold – the behavior of a CCA on a given configuration $\boldsymbol{x} \in \{0,1\}^N$ should depend only on whether the density is greater than or smaller than this threshold. The next definition singles out those CCAs fulfilling Definition 7.1.1, while not allowing the situation where $\rho(\boldsymbol{x}) < \frac{1}{2}$ but $\rho(\mathbb{F}(\boldsymbol{x})) > \frac{1}{2}$, or $\rho(x) > \frac{1}{2}$ but $\rho(\mathbb{F}(x)) < \frac{1}{2}$.

Definition 7.1.2 *Let $\mathbb{f}$ be the local rule of a CCA, $\mathbb{F}$ be the corresponding global rule and N be a given natural number. We say that this CCA solves the DCP* for N and we write $\mathbb{f} \in \mathrm{DCP}^*(N)$, if for any $\boldsymbol{x} \in \{0,1\}^N$ the following implications hold:*

$$\begin{aligned} \rho(\boldsymbol{x}) < \frac{1}{2} &\Rightarrow \left(\rho\left(\mathbb{F}(\boldsymbol{x})\right) \le \frac{1}{2} \quad \textit{and} \quad \lim_{t\to\infty} \mathbb{F}^t(\boldsymbol{x}) = 0^N\right) , \\ \rho(\boldsymbol{x}) > \frac{1}{2} &\Rightarrow \left(\rho\left(\mathbb{F}(\boldsymbol{x})\right) \ge \frac{1}{2} \quad \textit{and} \quad \lim_{t\to\infty} \mathbb{F}^t(\boldsymbol{x}) = 1^N\right) . \end{aligned} \tag{7.4}$$

Proposition 7.1.2 *In the case of binary CAs, the definition of* $\mathrm{DCP}(N)$ *is equivalent to the definition of* $\mathrm{DCP}^*(N)$.

Proof. It is obvious that if $\mathbb{f} \in \mathrm{DCP}^*(N)$, then $\mathbb{f} \in \mathrm{DCP}(N)$, since condition Eq. (7.4) is stronger than condition Eq. (7.3). Let us now assume that $\mathbb{f} \in \mathrm{DCP}(N)$. We need to show that if $\rho(\boldsymbol{x}) < \frac{1}{2}$, then $\rho(\mathbb{F}(\boldsymbol{x})) \le \frac{1}{2}$. Let $\boldsymbol{y} = \mathbb{F}(\boldsymbol{x})$. Assume that $\rho(\boldsymbol{y}) > \frac{1}{2}$. From $\mathbb{f} \in \mathrm{DCP}(N)$, we know that $\mathbb{F}^t(\boldsymbol{y}) \to 1^N$. On the other hand, $\mathbb{F}^t(\boldsymbol{x}) \to 0^N$, so also $\mathbb{F}^t(\mathbb{F}(\boldsymbol{x})) =$

$\mathbb{F}^t(\boldsymbol{y}) \to 0^N$, which leads to a contradiction. Therefore, indeed $\rho(\boldsymbol{y}) \leq \frac{1}{2}$. The case $\rho(\boldsymbol{x}) > \frac{1}{2}$ is similar. □

However, we do not know whether these definitions are equivalent in general. Since our goal is to find any local rule belonging to DCP(N), we use the fact that DCP*(N) ⊆ DCP(N) and concentrate on DCP*(N) because it is easier to study.

An easy computation shows that if l_i are the LUT entries and η, ϵ, γ are the parameters of a local rule $\mathbb{f}$, while $l_i^c, \eta^c, \epsilon^c, \gamma^c$ are their counterparts for its conjugate $\mathbb{f}^C$ (see Eq. (2.5)), then

$$\begin{aligned} l_i^c &= 1 - l_{7-i} \ , \\ \eta^c &= \epsilon \ , \\ \epsilon^c &= \eta, \gamma^c = -\gamma \ . \end{aligned}$$

As we explained before, the transformation described by Eq. (2.5) does not change the global dynamics of the corresponding CA. Hence, we get the following observation, which is very useful for simplifying proofs.

Corollary 7.1.3 *A local rule $\mathbb{f}$ belongs to* DCP*(N) *if and only if the conjugate rule $\mathbb{f}^C$ belongs to* DCP*(N).

7.2 Necessary conditions

We start with the well-known fact that if a local rule $\mathbb{f}$ solves the DCP and is continuous, then it must be double-quiescent. We give the proof only for the reader's convenience.

Lemma 7.2.1 *Let $\mathbb{f}$ be the local rule of a one-dimensional ACCA with unit radius. If $\mathbb{f} \in$ DCP*(N) for some $N \geq 1$, then $l_0 = 0$ and $l_7 = 1$.*

Proof. Let us introduce the auxiliary function $g\colon [0,1] \to [0,1]$ defined by $g(x) = f(x,x,x)$, then we need to show that $g(0) = 0$. For that purpose let us define the sequence: $c_0 = 0$ and $c_{t+1} = g(c_t)$ for $t \geq 0$. Then, we have for any t

$$\mathbb{F}^t(0^N) = c_t^N \ .$$

Since $\mathbb{F}^t(0^N) \xrightarrow{t\to\infty} 0^N$, we know that $c_t \xrightarrow{t\to\infty} 0$. Using the continuity of g, we get $g(c_t) \xrightarrow{t\to\infty} g(0)$ and as a result we have

$$g(0) = \lim_{t\to\infty} g(c_t) = \lim_{t\to\infty} c_{t+1} = \lim_{t\to\infty} c_t = 0 \ . \tag{7.5}$$

As $\mathbb{f} \in$ DCP*(N), the same holds for $\mathbb{f}^C$, hence $l_0^c = 0$, *i.e.*, $l_7 = 1$. □

Next, we want to obtain some dependencies among η, ϵ and γ. To do this, we introduce the following notation: for any natural number $k < N$, we denote by $d(N,k)$ the remainder of the division of N by k and let

$$d^+(N,k) = \begin{cases} d(N,k) & , \text{if } k \nmid N \\ k & , \text{if } k \mid N. \end{cases}$$

The following lemma allows us to reduce the huge set of all possible local rules solving the DCP to a smaller subset. To simplify the proof, we assume that $N \geq 20$, but similar estimates can be obtained for any N.

Lemma 7.2.2 *Let $\mathbb{f}$ be the local rule of a one-dimensional ACCA with unit radius and η, ϵ, γ be given by Eq. (7.1). If $\mathbb{f}$ belongs to $\mathrm{DCP}^*(N)$ for some $N \geq 20$, then the following inequalities hold:*

$$\frac{-2\epsilon - d^+(N,2)}{N - d^+(N,2) - 2} \leq \gamma \leq \frac{2\eta + d^+(N,2)}{N - d^+(N,2) - 2}\,, \tag{7.6}$$

$$\frac{4(\eta+\gamma) - 8 + 2d(N,4)}{N + 4 - d(N,4)} \leq \epsilon \leq \frac{4(\eta+\gamma) + 2d^+(N,4)}{N - d^+(N,4)}\,, \tag{7.7}$$

$$\frac{4(\epsilon-\gamma) - 8 + 2d(N,4)}{N + 4 - d(N,4)} \leq \eta \leq \frac{4(\epsilon-\gamma) + 2d^+(N,4)}{N - d^+(N,4)}\,. \tag{7.8}$$

Proof. Note that according to Lemma 7.2.1, the local rule $\mathbb{f}$ is double-quiescent. The justification of the above inequalities is based on choosing an appropriate configuration $\boldsymbol{x}$, calculating n_1, n_2, n_3 and n_7 (see Table 7.1) and computing $\rho(\mathbb{F}(\boldsymbol{x}))$ by applying the formula from Lemma 7.1.1. Then

$\boldsymbol{x}$	$\rho(\boldsymbol{x})$	n_1	n_2	n_3	n_7	N
$(01)^m 0^{d^+(N,2)}$	$< \frac{1}{2}$	1	m	0	0	$2m + d^+(N,2)$
$(011)^m 0^{m+d^+(N,4)}$	$< \frac{1}{2}$	1	0	m	0	$4m + d^+(N,4)$
$(011)^m 0^{m-4+d(N,4)}$	$> \frac{1}{2}$	1	0	m	0	$4(m-1) + d(N,4)$

Table 7.1: The number of neighborhoods N_1, N_2, N_3 and N_7 in the configurations considered in the proof of Lemma 7.1.1.

we can write $\rho(\mathbb{F}(\boldsymbol{x})) \leq \frac{1}{2}$ or $\rho(\mathbb{F}(\boldsymbol{x})) \geq \frac{1}{2}$, depending on whether $\rho(\boldsymbol{x}) < \frac{1}{2}$ or $\rho(\boldsymbol{x}) > \frac{1}{2}$.

To obtain Eq. (7.6), let us note that since $N \geq 20$, there is a natural number $m \geq 9$ such that $N = 2m + d^+(N,2)$. Hence, we can consider the configuration $\boldsymbol{x} = (01)^m 0^{d^+(N,2)}$. As we see in Table 7.1, for this configuration it holds that

$$\rho(\boldsymbol{x}) < \frac{1}{2} \text{ and } \rho(\mathbb{F}(\boldsymbol{x})) = \frac{1}{N}(m - \eta + \gamma(m-1))\,,$$

so we have

$$m - \eta + \gamma(m-1) \leq \frac{1}{2}\Big(2m + d^+(N,2)\Big)\,,$$

which yields

$$\gamma \leq \frac{2\eta + d^+(N,2)}{2m-2}\,.$$

Hence, we have just proven the right inequality in Eq. (7.6) as $2m = N - d^+(N,2)$. Considering now the conjugate rule $\mathbb{f}^C$, we get

$$\gamma^c \leq \frac{2\eta^c + d^+(N,2)}{N - d^+(N,2) - 2}$$

and substituting $\gamma^c = -\gamma, \eta^c = \epsilon$, we obtain the left inequality in Eq. (7.6).

Taking $N = 4m + d^+(N,4)$ for some natural number $m \geq 4$, we can apply the above procedure to the configuration $\boldsymbol{x} = (011)^m 0^{m+d^+(N,4)}$. In this

way, we get

$$2m - \eta - \gamma + \epsilon \cdot m \leq \frac{1}{2}(4m + d^+(N,4)) \ ,$$

hence

$$\epsilon \leq \frac{4(\eta+\gamma) + 2d^+(N,4)}{4m} = \frac{4(\eta+\gamma) + 2d^+(N,4)}{N - d^+(N,4)} \ ,$$

which is the right inequality in Eq. (7.7), while we get the right inequality in Eq. (7.8) considering the conjugate rule.

Finally, as $N = 4(m-1) + d(N,4)$ for some $m \geq 6$, we consider the configuration $\boldsymbol{x} = (011)^m 0^{m-4+d(N,4)}$. Hence, we get

$$2m - \eta - \gamma + \epsilon \cdot m \geq \frac{1}{2}(4(m-1) + d(N,4)) \ ,$$

which yields

$$\epsilon \geq \frac{4(\eta+\gamma) - 8 + 2d(N,4)}{4m} = \frac{4(\eta+\gamma) - 8 + 2d(N,4)}{N + 4 - d(N,4)} \ ,$$

i.e., the left inequality in Eq. (7.7) and using the conjugate rule, we get the left inequality in Eq. (7.8). □

Let us note that, since $\eta, \epsilon, \eta+\gamma, \epsilon-\gamma \in [-2,1]$ and for every $k < N$ we have

$$0 \leq d(N,k) \leq k-1 \text{ and } 1 \leq d^+(N,k) \leq k \ ,$$

the above lemma yields that if $\mathbb{f}$ belongs to $\mathrm{DCP}^*(N)$ for some $N \geq 20$, then

$$\begin{aligned} \gamma &\in \left[-\frac{4}{N-4}, \frac{4}{N-4}\right] \subset \left[-\frac{1}{4}, \frac{1}{4}\right] \ , \\ \eta, \epsilon &\in \left[-\frac{16}{N}, \frac{12}{N-4}\right] \subset \left[-\frac{4}{5}, \frac{3}{4}\right] \ . \end{aligned} \tag{7.9}$$

Hence, as a simple consequence we get an alternative proof of the following fact (see [2]).

Corollary 7.2.3 *There exists no one-dimensional ACCA with unit radius that solves the* $\mathrm{DCP}^*(N)$ *for every* N.

Proof. Contrary to our claim, let $\mathbb{f}$ be the local rule of an ACCA that belongs to $\mathrm{DCP}^*(N)$ for every N and let η, ϵ, γ be its parameters given by Eq. (7.1). Then, for Eq. (7.9), we have

$$\begin{aligned} \gamma &\in \bigcap_{N=20}^{\infty} \left[-\frac{4}{N-4}, \frac{4}{N-4}\right] = \{0\} \\ \eta, \epsilon &\in \bigcap_{N=20}^{\infty} \left[-\frac{16}{N}, \frac{12}{N-4}\right] = \{0\} \ , \end{aligned}$$

which means that $\eta = \epsilon = \gamma = 0$, but then the entries of the LUT of $\mathbb{f}$ satisfy Eq. (3.17), so $\mathbb{f}$ conserves density, which is impossible. □

The second consequence of Lemma 7.2.2 concerns ECAs. If the LUT of $\mathbb{f}$ contains only 0s and 1s, then the parameters η, ϵ, γ are integers, which implies $\eta = \epsilon = \gamma = 0$.

Corollary 7.2.4 *Let $N \geq 20$ be given. There exists no ECA that belongs to* DCP(N) *for a given N.*

To further reduce the set of local rules that will be considered as potential solutions of DCP, we will show that if,

$$l_1, l_2, \ldots, l_6 \in (0, 1) \ ,$$

then

$$l_1 + l_2 + l_4 \leq 1 \ ,$$

while

$$l_3 + l_5 + l_6 \geq 2 \ .$$

In our argumentation, the following observation plays a key role.

Lemma 7.2.5 *Let $\mathbb{f}$ be the local rule of a one-dimensional ACCA with unit radius and with LUT $(l_0, \ldots, l_7)$. If $l_1, l_2, \ldots, l_6 \in (0, 1)$, then for every binary configuration $\boldsymbol{x} \in \{0,1\}^N \setminus \{0^N, 1^N\}$ and for any $t \geq N$, the configuration $\mathbb{F}^t(\boldsymbol{x})$ contains neither* 0s *nor* 1s*.*

Proof. As the local rule $\mathbb{f}$ is affine in each variable, the assumption $l_1, l_2, \ldots, l_6 \in (0, 1)$ guarantees that

$$\mathbb{f}(x, y, z) = 0 \iff x = y = z = 0$$

and

$$\mathbb{f}(x, y, z) = 1 \iff x = y = z = 1 \ .$$

This means that if $\boldsymbol{x}$ is different from 0^N and 1^N, the length of blocks of 0s and 1s reduces by at least one in each transition. Moreover, no new 0s and 1s appear. □

Lemma 7.2.6 *Let $\mathbb{f}$ be the local rule of an ACCA with unit radius and with LUT $(l_0, \ldots, l_7)$ such that $l_1, l_2, \ldots, l_6 \in (0, 1)$. If $\mathbb{f}$ belongs to* DCP$^*(N)$ *for some $N \geq 3$, then $\eta \geq 0$ and $\epsilon \geq 0$, i.e.*

$$l_1 + l_2 + l_4 \leq 1 \quad \textit{and} \quad l_3 + l_5 + l_6 \geq 2 \ .$$

Proof. Suppose, contrary to our claim, that $\eta < 0$. Since from Eq. (3.4) we obtain that

$$\mathbb{f}'_x(0, 0, 0) = l_4 > 0 \ ,$$
$$\mathbb{f}'_y(0, 0, 0) = l_2 > 0 \ ,$$
$$\mathbb{f}'_z(0, 0, 0) = l_1 > 0 \ ,$$

according to the continuity of $\mathbb{f}'_x$, $\mathbb{f}'_y$ and $\mathbb{f}'_z$, there exists a $\delta > 0$ such that

for all $x, y, z \in [0, \delta]$ it holds that

$$\begin{aligned} \mathbb{f}'_x(x, y, z) &> 0 \ , \\ \mathbb{f}'_y(x, y, z) &> 0 \ , \\ \mathbb{f}'_z(x, y, z) &> 0 \ . \end{aligned}$$

Consequently, as $\mathbb{f}$ is affine in each variable, we get from Theorem 3.1.1 that, for any $m \leq \delta$, if $x, y, z \in [m, \delta]$, then

$$\mathbb{f}(x, y, z) \geq \mathbb{f}(m, m, m) \ .$$

Moreover, for $m \in [0, -\eta/16] \subset [0, 1]$, we have that

$$\mathbb{f}(m, m, m) > (1 - \eta/2)m \ .$$

Indeed, one can verify that

$$\begin{aligned} \mathbb{f}(m, m, m) &= m\left[-(\epsilon + \eta)m^2 + (\epsilon + 2\eta)m + (1 - \eta)\right] \\ &\geq m(-2m^2 - 6m + 1 - \eta) > m(1 - \eta - 8m) \\ &\geq m\left(1 - \eta - 8 \cdot \frac{-\eta}{16}\right) = (1 - \eta/2)m \ . \end{aligned}$$

Combining these facts, we know that for each $m \leq \delta_0$ and $x, y, z \in [m, \delta_0]$

$$\mathbb{f}(x, y, z) > (1 - \eta/2)m \ , \tag{7.10}$$

where $\delta_0 = \min(\delta, -\eta/16)$.

Now let $\boldsymbol{x}$ be any configuration satisfying $0 < \rho(\boldsymbol{x}) < \frac{1}{2}$. Since $\mathbb{F}^t(\boldsymbol{x}) \to 0^N$ if $t \to \infty$, there exists a $T > N$ such that for $t \geq T$

$$\mathbb{F}^t(\boldsymbol{x}) \in [0, \delta_0]^N \ . \tag{7.11}$$

Let $m_T = \min\left\{x_0^T, x_1^T, \ldots, x_{N-1}^T\right\}$. From Lemma 7.2.5 it follows that $m_T > 0$ and, of course, $m_T \leq \delta_0$. From Eq. (7.10), we conclude that for any $i \in \{0, 1, \ldots, N-1\}$, it holds that

$$x_i^{N+1} > \left(1 - \frac{1}{2}\eta\right) m \ .$$

By induction, for any natural number n, we have

$$x_i^{N+n} > \left(1 - \frac{1}{2}\eta\right)^n m \ .$$

But this contradicts Eq. (7.11) as

$$1 - \frac{1}{2}\eta > 1 \ .$$

We conclude the proof that $\eta \geq 0$. The fact that $\epsilon \geq 0$ can be obtained using the conjugate rule. □

Taking into account considerations from Lemma 7.2.6 and Lemma 7.2.2, we get the following corollary.

Corollary 7.2.7 *Let $\mathbb{f}$ be the local rule of a one-dimensional ACCA with unit radius and with LUT $(l_0, \ldots, l_7)$ such that $l_1, l_2, \ldots, l_6 \in (0,1)$. If $\mathbb{f}$ belongs to* $\mathrm{DCP}^*(N)$ *for some $N \geq 20$, then the following inequalities hold*

$$\frac{-2\epsilon - d^+(N,2)}{N - d^+(N,2) - 2} \leq \gamma \leq \frac{2\eta + d^+(N,2)}{N - d^+(N,2) - 2}\,, \tag{7.12}$$

$$0 \leq \epsilon \leq \frac{4(\eta+\gamma) + 2d^+(N,4)}{N - d^+(N,4)}\,, \tag{7.13}$$

$$0 \leq \eta \leq \frac{4(\epsilon-\gamma) + 2d^+(N,4)}{N - d^+(N,4)}\,. \tag{7.14}$$

Figure 7.2 illustrates that the solid S (black) containing all points $(\eta, \varepsilon, \gamma)$ fulfilling Eqs. (7.12)–(7.14) for $N = 23$ is considerably smaller than the cuboid (grey) consisting of all points $(\eta, \varepsilon, \gamma)$ fulfilling $\gamma \in [-1, 1]$ and $\eta, \varepsilon \in [-2, 1]$.

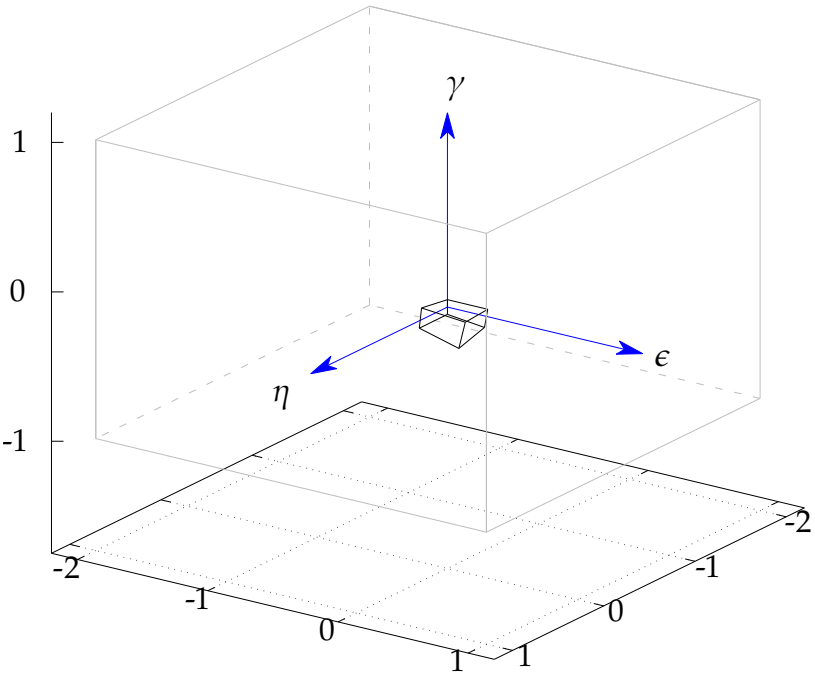


Figure 7.2: The solid S (black), *i.e.*, the set of points (η, ϵ, γ) whose coordinates satisfy Eqs. (7.12)–(7.14) versus the set of points (η, ϵ, γ) for which $\gamma \in [-1, 1]$ and $\eta, \epsilon \in [-2, 1]$ (grey).

7.3 Sufficient stopping conditions

The following theorem reduces the waiting time to get a response to the question whether or not the considered local rule belongs to $\mathrm{DCP}^*(N)$.

Theorem 7.3.1 *Let $\mathbb{f}$ be the local rule of a one-dimensional ACCA with unit radius and with LUT $(l_0, \ldots, l_7)$ such that $l_1, l_2, \ldots, l_6 \in (0,1)$. Assume that $\mathbb{f}$ is double-quiescent and $\eta, \epsilon \in (0,1)$. Let $s_0 = \frac{\eta}{2(\eta+\epsilon)}, s_1 = \frac{\epsilon}{2(\eta+\epsilon)}$ and*

$$q = \max\left\{l_1, \ldots, l_6, 1 - l_1, \ldots, 1 - l_6, 1 - \frac{1}{4}\eta, 1 - \frac{1}{4}\epsilon\right\}\,.$$

Then the following implications hold

$$\Big(\forall\, s \in [0, s_0]\Big)\Big(\boldsymbol{x} \in [0, s]^N \Rightarrow \mathbb{F}(\boldsymbol{x}) \in [0, q \cdot s]^N\Big)\,, \tag{7.15}$$

$$\Big(\forall\, s \in [0, s_1]\Big)\Big(\boldsymbol{x} \in [1-s, 1]^N \Rightarrow \mathbb{F}(\boldsymbol{x}) \in [1-q \cdot s, 1]^N\Big)\,. \tag{7.16}$$

Proof. To prove Eq. (7.15), let us assume that $\boldsymbol{x} \in [0, s]^N$, where $0 \le s \le s_0$. From Theorem 3.1.1, it follows that for each $i \in \{0, \ldots, N-1\}$ it holds that

$$\mathbb{f}(x_{i-1}, x_i, x_{i+1}) \le \max\left\{\mathbb{f}(\boldsymbol{y}) \mid \boldsymbol{y} \in \{0, s\}^3\right\}\,.$$

Thus, if we prove that each of the eight elements of the set

$$\left\{\mathbb{f}(\boldsymbol{y}) \mid \boldsymbol{y} \in \{0, s\}^3\right\}$$

is not greater than $q \cdot s$, we showed our claim.

From Eq. (3.4) we easily get that

$$\begin{aligned}
\mathbb{f}(0,0,0) &= 0\,,\\
\mathbb{f}(0,0,s) &= l_1 \cdot s\,,\\
\mathbb{f}(0,s,0) &= l_2 \cdot s\,,\\
\mathbb{f}(s,0,0) &= l_4 \cdot s\,,
\end{aligned}$$

which all are not greater than $q \cdot s$. Furthermore, since

$$\begin{aligned}
l_3 &\le q\,,\\
l_1 + l_2 \le l_1 + l_2 + l_4 = 1 - \eta &\le 1 - \frac{1}{4}\eta = q\,,
\end{aligned}$$

it holds that

$$\begin{aligned}
\mathbb{f}(0,s,s) &= \left((l_3 - l_2 - l_1)s + l_2 + l_1\right) s = \left(l_3 s + (l_2 + l_1)(1-s)\right) s\\
&\le \left(qs + q(1-s)\right) s = q \cdot s\,.
\end{aligned}$$

Similarly, we get

$$\begin{aligned}
\mathbb{f}(s,0,s) &= \left((l_5 - l_4 - l_1)s + l_4 + l_1\right) s = \left(l_5 s + (l_4 + l_1)(1-s)\right) s\\
&\le \left(qs + q(1-s)\right) s = q \cdot s\,,
\end{aligned}$$

and

$$\begin{aligned}
\mathbb{f}(s,s,0) &= \left((l_6 - l_4 - l_2)s + l_4 + l_2\right) s = \left(l_6 s + (l_4 + l_2)(1-s)\right) s\\
&\le \left(qs + q(1-s)\right) s = q \cdot s\,.
\end{aligned}$$

For the value

$$\mathbb{f}(s,s,s) = g(s) = \left(-(\epsilon+\eta)s^2 + (\epsilon + 2\eta)s + (1-\eta)\right) s\,,$$

the argumentation is a little more complicated. First, let us note that the quadratic function

$$h(s) = -(\epsilon+\eta)s^2 + (\epsilon + 2\eta)s + (1-\eta)$$

is increasing on $[0, s_0]$, as it increases on $\left(-\infty, \frac{\epsilon+2\eta}{2(\eta+\epsilon)}\right]$ with $s_0 \le \frac{\epsilon+2\eta}{2(\eta+\epsilon)}$. It

then follows that:

$$\mathbb{f}(s,s,s) = h(s)\cdot s \leq h(s_0)\cdot s = \left(1-\frac{2\epsilon\eta+\eta^2}{4(\eta+\epsilon)}\right)s \leq \left(1-\frac{1}{4}\eta\right)s \leq q\cdot s\ ,$$

which concludes the proof of Eq. (7.15). Now, to argue Eq. (7.16), it suffices to use the conjugate rule and note that

$$q^c = \max\left\{l_1^c,\ldots,l_6^c,1-l_1^c,\ldots,1-l_6^c,1-\frac{1}{4}\eta^c,1-\frac{1}{4}\epsilon^c\right\} = q\ .$$

□

Note that we have just proved that if the states of all cells of a configuration $\boldsymbol{x}$ are small enough, then $\mathbb{F}^t(\boldsymbol{x}) \to 0^N$ if $t \to \infty$, whereas we have that $\mathbb{F}^t(\boldsymbol{x}) \to 1^N$ if all cells have values that are great enough. Moreover, in both cases, the convergence is exponential with base q.

Summarizing, we have proved some facts that will help us to find local rules belonging to DCP*(N) for a given N. We are going to check for every possible local rule whether it belongs to DCP*(N). As the set of possible local rules is huge, we first give some necessary conditions for a local rule to belong to DCP*(N). This allows us to concentrate on a fairly small subset (see Corollary 7.2.7). Secondly, we will use the condition that allows us to decide whether a given local rule belongs to DCP*(N) (see Theorem 7.3.1).

7.4 Simulation results

In this section, we present the results obtained for ACCAs solving the DCP* for $N = 23$. We enumerated the set of double-quiescent local rules by their LUTs $(0, l_1, \ldots, l_6, 1)$ and we considered only the cases where every l_i was of the form $0.02k$, $k = 1,\ldots,49$. For each LUT, we computed the parameters η, ϵ, γ given by Eq. (3.11) and we verified the necessary conditions in Eqs. (7.12)–(7.14). If these equations were not satisfied, then the local rule was omitted from further considerations, because Corollary 7.2.7 states that this rule cannot belong to DCP*(23). Otherwise, the rule $\mathbb{f}$ was tested on the set of all binary configurations $\mathcal{I}_{23} = \{0,1\}^{23}$. For each configuration $\boldsymbol{x} \in \mathcal{I}_{23}$ and for each consecutive iteration $\mathbb{F}^t(\boldsymbol{x})$, we checked whether all values x_i^t were smaller than $\frac{\eta}{2(\epsilon+\eta)}$ (resp. greater than $\frac{\epsilon}{2(\epsilon+\eta)}$). If one of these conditions was satisfied, then we stopped further computations for configuration $\boldsymbol{x}$, since Theorem 7.3.1 assures that $\mathbb{F}^t(\boldsymbol{x})$ converges to 0^N (resp. to 1^N) with $t \to \infty$. The time step t at which one of these conditions was satisfied, is denoted as $T_f(\boldsymbol{x})$. Next, we checked whether the considered local rule correctly classified the density of the configuration $\boldsymbol{x}$ or not. If so, we assigned $\boldsymbol{x}$ to the set $P(\mathbb{f})$ of correctly classified configurations, otherwise to the set $M(\mathbb{f})$ of misclassified configurations. If the answer was not reached within $\Lambda = 2000$ iterations, the simulation was halted and the configuration $\boldsymbol{x}$ was assigned to the set $U(\mathbb{f})$ of unknown cases. Of course, if $M(\mathbb{f}) \neq \emptyset$, then the local rule $\mathbb{f}$ does not belong to DCP*(23), so we were interested only in these LUTs (these $\mathbb{f}$) for which $M(\mathbb{f}) = \emptyset$. It appeared that in all those cases also the set $U(\mathbb{f})$ was empty. Finally, for local rules belonging

to DCP*(23), we calculated the worst time $T_{\mathbb{f}}$ as the maximum over all $T_{\mathbb{f}}(\boldsymbol{x})$. There were 102 299 such rules, but when we checked the parameters η, ϵ, γ for these rules, it turned out that, for our experimental set-up, there were only nine unique possibilities, which are presented in Table 7.2. The location of these points $(\eta, \varepsilon, \gamma)$ in the solid S is shown in Figure 7.3.

Table 7.2: Parameters η, ϵ, γ of the local rules that solve the DCP* for $N = 23$.

η	ϵ	γ	$case$	n_{good}	n_{total}	% of rules solving DCP	T_{min}
0.02	0.02	−0.04	Case 1	126	33456	0.4%	622
0.02	0.02	−0.02	Case 2	18505	34586	53.5%	350
0.02	0.02	0	Case 3	34870	35720	97.6%	302
0.02	0.02	0.02	Case $(2)^C$	18500	34580	53.5%	350
0.02	0.02	0.04	Case $(1)^C$	126	33456	0.4%	622
0.04	0.04	−0.02	Case 4	4222	32429	13.0%	222
0.04	0.04	0	Case 5	21558	33511	64.3%	186
0.04	0.04	0.02	Case $(4)^C$	4222	32423	13.0%	222
0.06	0.06	0	Case 6	170	31395	0.5%	195

For each unique (η, ϵ, γ) in Table 7.2, we counted how many rules from

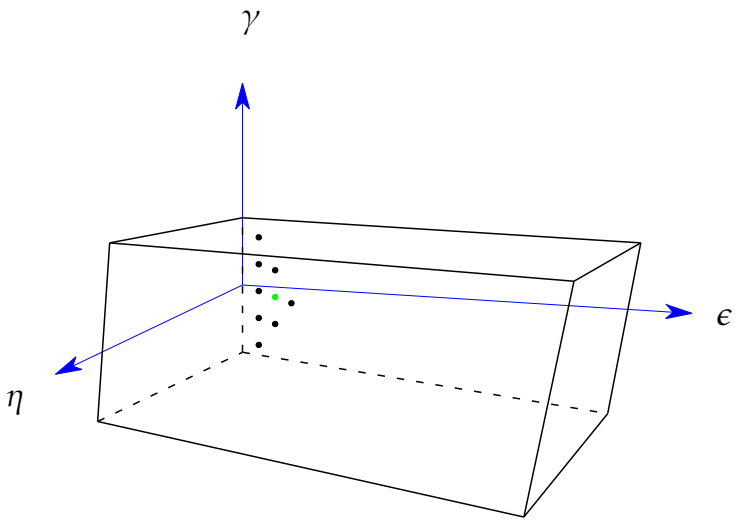


Figure 7.3: The location of the points $(\eta, \varepsilon, \gamma)$ listed in Table 7.2 in the solid S. The green point corresponds to Case 5 (as given in Table 7.2).

the considered ones have this set of parameters (n_{total}) and how many of them belong to DCP*(23) (n_{good}) and the minimal time T_{min} over all $T_{\mathbb{f}}$ for those rules. Moreover, according to the transformation described by Eq. (2.5), some cases are mutually conjugate, so finally there are only six different cases. Figure 7.4 presents each of the above cases separately. Every plot shows a black polyhedron consisting of all local rules having (η, ϵ, γ) as parameters and the location of the points corresponding to the local rules which belong to DCP*(23). We are interested in highly performant rules, *i.e.*, rules with the lowest $T_{\mathbb{f}}$. Let us note that we can compare $T_{\mathbb{f}}$ for rules from different cases in Table 7.2, since in each of these cases $\eta = \varepsilon$, so we always have $s_0 = s_1 = \frac{1}{4}$. It turned out that there are two mutually conjugate rules (both from Case 5) leading to the lowest $T_{\mathbb{f}} = 186$. We present the LUTs of these rules in Table 7.3. Surprisingly the LUTs of the nearest to $\mathbb{f}_{best}$ or $\mathbb{f}^C_{best}$ (using the Euclidean

Table 7.3: The most performant rules for $N = 23$.

	a	b	c	l_0	l_1	l_2	l_3	l_4	l_5	l_6	l_7
$\mathbb{f}_{best}$	0.12	0.12	0.02	0.00	0.12	0.02	0.88	0.82	0.98	0.18	1.00
$\mathbb{f}^C_{best}$	0.82	0.82	0.02	0.00	0.82	0.02	0.18	0.12	0.98	0.88	1.00

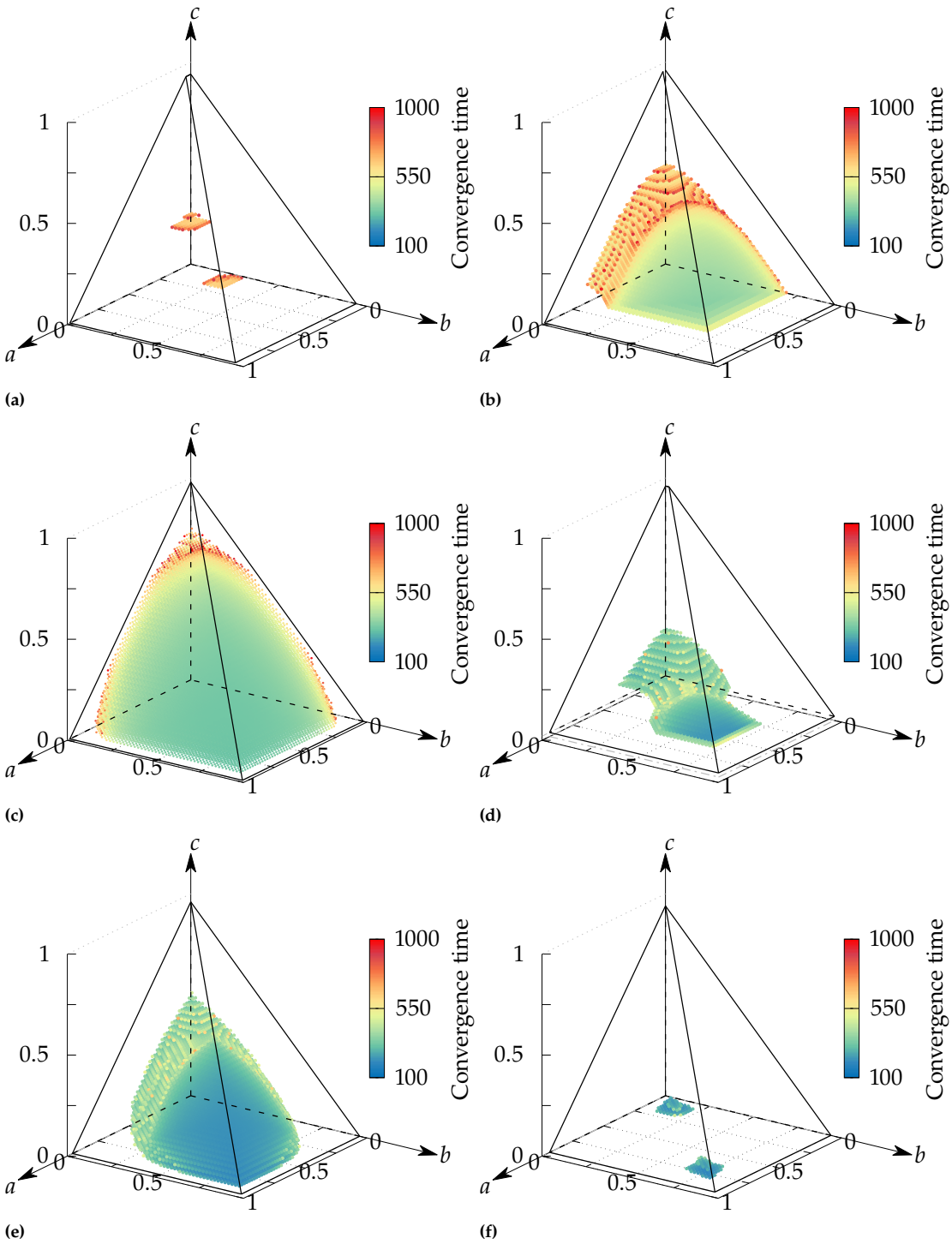


Figure 7.4: The location of the points (a, b, c) corresponding to the local rules that belong to DCP*(23) (coloured area) in the polyhedron enclosing all local rules, as defined by the parameters (η, ϵ, γ) listed in Table 7.2.

metric in the space $[0,1]^8$) are $(0,0,0,1,1,1,0,1)$ and $(0,1,0,0,0,1,1,1)$, *i.e.*, the traffic-right rule (ECA184) and the traffic-left rule (ECA226). The positions of $\mathbb{f}_{best}$ and $\mathbb{f}^{C}_{best}$ in the polyhedron for Case 5 are shown in Figure 7.5, while Figure 7.6 shows the dynamics of $\mathbb{f}_{best}$ starting from the configuration $\boldsymbol{x} = 0^9 1^2 (0111)^3$, which has the worst classification time in terms of the maximal value (shown in blue) and the average value (shown in green) of the cell states $\{x_0^t, x_1^t, \ldots, x_{22}^t\}$. The evolution of $\mathbb{f}_{best}$

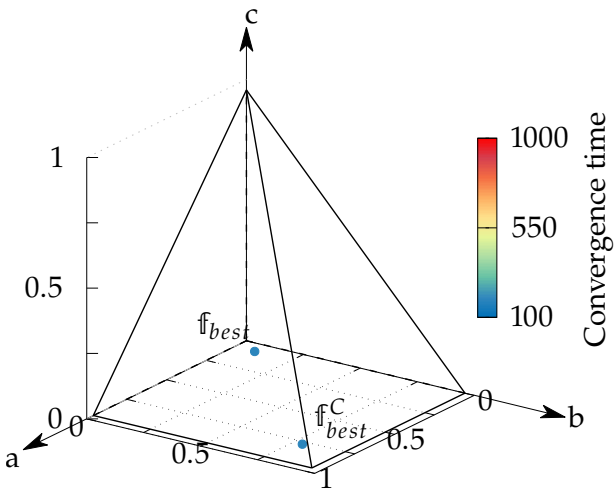


Figure 7.5: The location of the points corresponding to $\mathbb{f}_{best}$ and $\mathbb{f}^C_{best}$ in the pyramid for Case 5.

for this configuration is quite simple. The cells tend to align their states, while they also evolve to 0 (as $\rho(\boldsymbol{x}) < \frac{1}{2}$). In Figure 7.6 we also compare

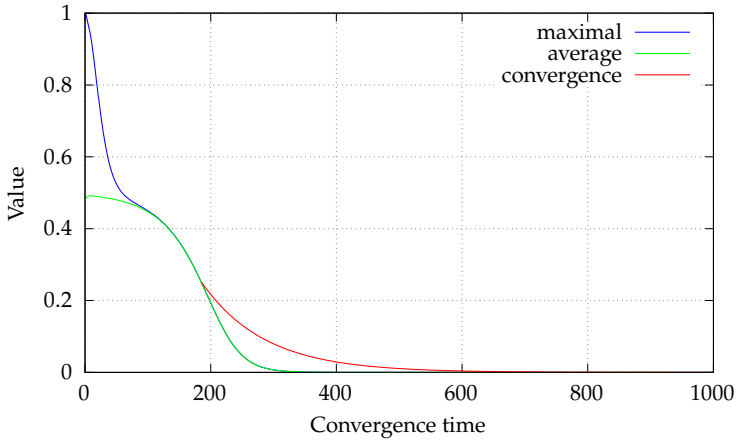


Figure 7.6: The maximal value (blue) and average value (green) of the cell states and the theoretical convergence limit (red), for the evolution of $\mathbb{f}_{best}$ on the initial configuration $\boldsymbol{x} = 0^9 1^2 (0111)^3$.

our results with the knowledge from Theorem 7.3.1: the red line starts for $t = T_{\mathbb{f}_{best}}(\boldsymbol{x})$ at the height $s_0 = \frac{1}{4}$ and exponentially decays to 0 with base $q = 0.99$. As Theorem 7.3.1 asserts, the blue line is under the red one.

7.5 Conclusions

The fixed-length DCP was discussed in the context of ACCAs. As the results of our computational simulations suggest, the solution sets have a non-trivial structure and contain rules greatly differing in their performance (convergence time). Using the theoretical results presented in Section 7.1, the simulations were done only for $N = 23$, but can be easily repeated for different N. The results presented in this chapter are important in the context of theoretical studies on the computational abilities of local-only computational models.

Solving the Relaxed Two-Dimensional DCP

8

8.1 Experimental set-up . . . 97
8.2 Experiments for a 5×5 grid 98
8.3 Experiments for a 23×23 grid 100
8.4 Conclusions 102

In Chapter 6, we analyzed the behavior of ACCAs in the context of the relaxed DCP in the one-dimensional case and we found that the best-performing ACCA was a slightly modified version of a traffic rule. Moreover, it performed about two times faster than the well-known averaging rule. In this chapter, we consider two-dimensional ACCAs, which incur a very high computational complexity [75]. For that reason, examining which two-dimensional density-conserving ACCAs are performing the best is much more challenging. The answer to this question is interesting because in two dimensions there is no rule that acts like the traffic rule in one dimension [76]. In the case of two-dimensional ACCAs, as the computational complexity is very high, we cannot consider all the simulations that we conducted in the one-dimensional case. Instead, we concentrate on finding rules that are better than the averaging rule $\mathbb{f}_d$ defined by Eq. (2.8).

8.1 Experimental set-up

For a given square grid $\mathcal{C}$ (with $n = m$ an odd number), we first selected a set of binary initial configurations $\mathcal{I}_\mathcal{C} \subset \mathcal{X}_n$, where $\mathcal{X}_n$ contains all configurations $\boldsymbol{x} \in \mathcal{X}$ such that $\rho(\boldsymbol{x}) = \frac{n^2-1}{2n^2}$. This was done because the difficulty of the relaxed DCP increases as the density of the initial configuration gets closer to $\frac{1}{2}$. We measured the convergence time T_0, defined as the minimum T that realizes Eq. (4.3) for the diffusion ACCA ($\mathbb{f}_d$). As the set $\mathcal{X}_n$ may be huge, this was done only for all $\boldsymbol{x} \in \mathcal{I}_\mathcal{C}$.

Next, we sampled the set of all density-conserving two-dimensional ACCAs, parameterized by (a, b, c, d, e, f, g, h), from the polytope P shown in Figure 3.5. This was done by varying every parameter with a step size of $\frac{1}{20}$. For every such parameterization, we then defined the local rule $\mathbb{f}$ by its LUT presented in Eq. (3.17), which led to 8 049 992 different rules all together. For each of them, we measured the convergence time $T_\mathbb{f}$ in the following way. For every $\boldsymbol{x} \in \mathcal{I}_\mathcal{C}$, we measured the minimum T for which all x_i^T were smaller than $\frac{1}{2}$ and denoted it by $T_\mathbb{f}(\boldsymbol{x})$. If for some $\boldsymbol{x} \in \mathcal{I}_\mathcal{C}$ this time $T_\mathbb{f}(\boldsymbol{x})$ was greater than or equal to T_0, then the local rule $\mathbb{f}$ is not better than the averaging rule, so we stopped the simulations after $\Lambda = T_0 + 1$ iterations. If for every $\boldsymbol{x} \in \mathcal{I}_\mathcal{C}$ the time $T_\mathbb{f}(\boldsymbol{x})$ was smaller than T_0, then we defined the convergence time as the maximum $T_\mathbb{f}(\boldsymbol{x})$. The fact that we ran the experiments on the set $\mathcal{I}_\mathcal{C}$, ignoring for example the configurations with density $\frac{n^2+1}{2n^2}$, significantly reduced the computation time. Yet, doing so makes that a rule with a performance better than the averaging rule that was retrieved on the basis of the set $\mathcal{I}_\mathcal{C}$ will not necessarily perform as such for other initial configurations, and hence might not be better than the averaging rule $\mathbb{f}_d$. To ensure that a rule will also be better than the averaging rule on other initial configurations, we have to match 16 symmetric rules, with the same convergence time, *i.e.*, vertical, horizontal, left and right diagonal, $90°, 180°, 270°$ rotations for the rule $\mathbb{f}$ and their conjugates.

When we found all rules that perform better than $\mathbb{f}_d$, we chose one of them (denoted by $\mathbb{f}_\star$) and ran a more detailed experiment on the set $\mathcal{I}_C$ or – if the set $\mathcal{I}_C$ seemed too small – on the set $\mathcal{I}_C^*$, where $\mathcal{I}_C \subset \mathcal{I}_C^* \subset \mathcal{X}_n$. For every configuration $\boldsymbol{x} \in \mathcal{I}_C^\star$ we measured the difference

$$\Delta T(\boldsymbol{x}) = T_{\mathbb{f}_d}(\boldsymbol{x}) - T_{\mathbb{f}}(\boldsymbol{x}) \ ,$$

to see how much better or worse the rule $\mathbb{f}_*$ performs compared to the averaging rule $\mathbb{f}_d$ (Eq. (2.8)) on a given configuration. As a result, we get a detailed insight into how much better the performance of a rule $\mathbb{f}_*$ is for a given set of initial configurations.

Additionally, we conducted a detailed study of two special kinds of density-conserving ACCAs that we defined earlier. The first were the rotation-symmetric ones, where the set of these CAs is parameterized by $\alpha \in [0, \frac{1}{4}]$ (Section 3.7, Eq. (3.22)). We sampled this range with a step of $\frac{1}{1000}$ – due to some singularity we observed at $\alpha = \frac{1}{4}$. The second kind of density-conserving ACCAs we considered were the CAs whose local rule does not depend on y and v (Subsection 3.6.4, Eq. (3.19)). For this experiment, we varied the parameters b and d with a step size of $\frac{1}{100}$, which resulted in 5151 rules.

8.2 Experiments for a 5×5 grid

In this experiment, we set $n = 5$. The set $\mathcal{I}_C$ contained all possible initial configurations satisfying the experimental constraint, *i.e.*, $\mathcal{I}_C = \mathcal{X}_n$, so $|\mathcal{I}_C| = 208012$. Since we found in this case that $T_0 = 12$, we halted computations after $\Lambda = 13$ iterations. Among the 8 049 992 investigated rules, we found 200 that were performing better than the averaging rule $\mathbb{f}_d$. Taking into account symmetries, we obtained the three independent rules $\mathbb{f}_1, \mathbb{f}_2, \mathbb{f}_3$ presented in Table 8.1, which all solve the relaxed DCP in $T = 10$ iterations (*i.e.*, two iterations faster than the averaging rule).

Table 8.1: Parameters of the best-performing rules found for a 5×5 grid.

rule	a	b	c	d	e	f	g	h
$\mathbb{f}_1$	0.20	0.20	0.40	0.10	0.35	0.25	0.25	0.45
$\mathbb{f}_2$	0.25	0.30	0.50	0.10	0.35	0.40	0.15	0.40
$\mathbb{f}_3$	0.30	0.25	0.50	0.10	0.35	0.40	0.20	0.55

For example, according to Eq. (3.18), the rule $\mathbb{f}_1$ is defined by:

$$\mathbb{f}_1\left(\begin{array}{ccc} & x & \\ y & z & u \\ & v & \end{array}\right) = \frac{1}{20}(v - u + y - x)z + \frac{1}{4}(x + y) + \frac{1}{5}(u + v) + \frac{1}{10}z \ .$$

Although we only checked that this rule is better than the averaging rule on the initial configurations from $\mathcal{I}_C$, due to symmetry and conjugation we know that it is better on all configurations with density $\frac{12}{25}$ and $\frac{13}{25}$. So, we conjecture that it is better for any initial configuration because for a 5×5 grid the densities $\frac{12}{25}$ and $\frac{13}{25}$ are the closest ones possible to $\frac{1}{2}$.

It should be emphasized that if for some local rule $\mathbb{f}$ we have $T_{\mathbb{f}_d} > T_{\mathbb{f}}$, then this does not exclude a situation that for a certain configuration $\boldsymbol{x} \in \mathcal{I}_C$ we have $T_{\mathbb{f}_d}(\boldsymbol{x}) < T_{\mathbb{f}}(\boldsymbol{x})$. To perform a detailed study on this matter, we took rule $\mathbb{f}_1$ as $\mathbb{f}_*$ and for each configuration $\boldsymbol{x} \in \mathcal{I}_C$ we computed the difference

$\Delta T(\boldsymbol{x})$ to compare how fast these rules act on every configuration in $\mathcal{I}_C$ (Figure 8.1). We found that both rules give the same result for 39.1% of the initial configurations, while in 50% of the cases, the rule $\mathbb{f}_\star$ reaches convergence up to two iterations faster.

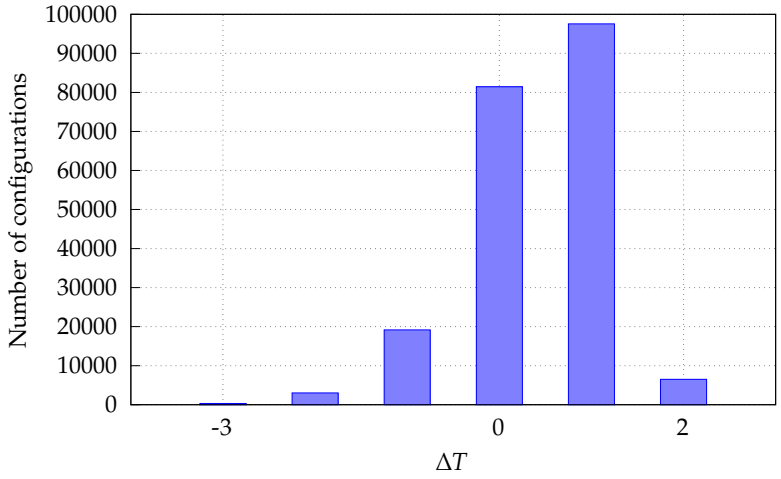
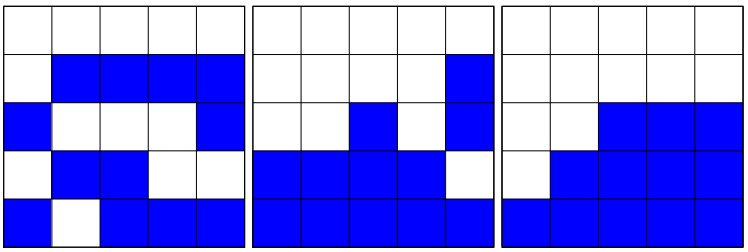


Figure 8.1: The number of configurations $\boldsymbol{x} \in \mathcal{I}_C$ with a given time difference $\Delta T(\boldsymbol{x})$ between rule $\mathbb{f}_d$ and $\mathbb{f}_\star$ for a 5×5 grid.

Examples of randomly chosen configurations $\boldsymbol{x} \in \mathcal{I}_C$ for some given values of $\Delta T(\boldsymbol{x})$ are presented in Figure 8.2.



(a) $\Delta T(\boldsymbol{x}) = -3$ **(b)** $\Delta T(\boldsymbol{x}) = 0$ **(c)** $\Delta T(\boldsymbol{x}) = 2$

Figure 8.2: Configurations $\boldsymbol{x}$ for given values of $\Delta T(\boldsymbol{x})$ and a 5×5 grid.

Additionally, we checked which of the rotation-symmetric, density-conserving ACCAs is the fastest one. The best-performing rules solving the relaxed DCP in $T = 11$ iterations have $\alpha \in (0.207, 0.237)$. As these are manifold, we picked those for which the number of configurations with the same number of iterations $T = 11$ required to solve the relaxed DCP is the lowest one. As a result we got only one rule for $\alpha = 0.228$. The results of this experiment are presented in Figure 8.3. We observe an interesting singularity: above a specific value of α the convergence time increases again. This is visible in the inset in Figure 8.3, where this happens for $\alpha = 0.228$. If we increase α, then first the number of configurations with convergence time $T = 11$ increases and then the convergence time itself. This experiment showed that among the rotation-symmetric ACCAs, we can find rules that are better than the averaging rule, with a convergence time that is only slightly higher than the best possible one.

Finally, we considered the rules given by Eq. (3.19) and found that only those with parameters $b = 0$ or $d = 0$ or $b + d = 1$ do not solve the relaxed DCP (*i.e.*, those placed on the edges). The lowest possible convergence time equals $T = 21$ and we found only three rules parameterized by

$$(b, d) \in \left\{ \left(\frac{8}{25}, \frac{8}{25} \right), \left(\frac{9}{25}, \frac{8}{25} \right), \left(\frac{8}{25}, \frac{9}{25} \right) \right\}$$

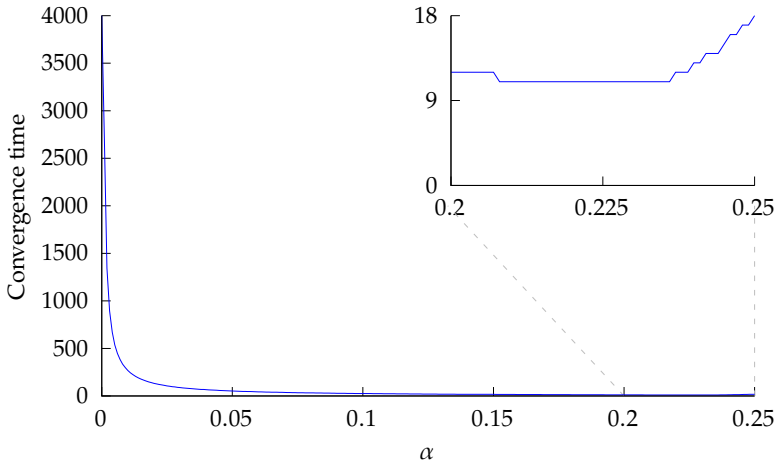


Figure 8.3: Results for a 5×5 grid and rotation-symmetric ACCAs.

that solve the relaxed DCP with this convergence time. Details for this experiment are presented in Figure 8.4. The results confirm our

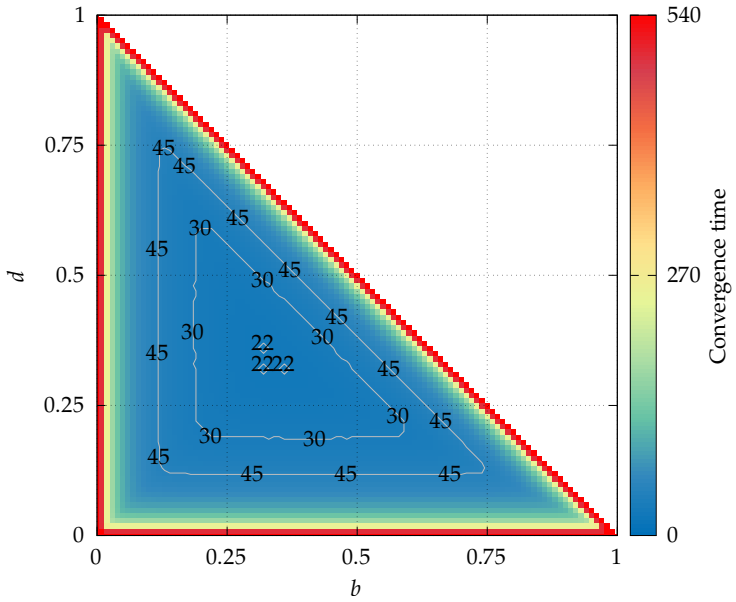


Figure 8.4: Results for density-conserving ACCAs whose local rule is independent of y and v for a 5×5 grid .

expectations that among the ACCAs whose local rule is independent of y and v, there is no rule with a low convergence time.

8.3 Experiments for a 23×23 grid

In this experiment, we set $n = 23$. As set $\mathcal{I}_C$ we now chose only a small subset of $\mathcal{X}_{23}$ ($|\mathcal{I}_C| = 25000$). We found $T_0 = 439$, so we halted the computations after $\Lambda = 440$ iterations. As a result, we got 66 312 rules that might be performing better than the averaging rule. Thirty-two of them might solve the relaxed DCP in the lowest number of iterations $T = 345$ and correspond to two independent rules (Table 8.2).

As mentioned before, investigating two-dimensional ACCAs incurs a very high computational cost. For that reason, we cannot experimentally verify whether such rules are truly better than the averaging rule or only for the initial configurations in $\mathcal{I}_C$.

rule	a	b	c	d	e	f	g	h
$\mathbb{f}_4$	0.40	0.05	0.25	0.05	0.45	0.55	0.45	0.70
$\mathbb{f}_5$	0.45	0.05	0.70	0.05	0.10	0.05	0.40	0.65

Table 8.2: Parameters of the best-performing rules found for a 23×23 grid.

To conduct more detailed simulations, we took rule $\mathbb{f}_4$ as $\mathbb{f}_\star$ and computed the convergence time for $\mathbb{f}_\star$ once again on a larger set of initial configurations $\mathcal{I}_C^\star \subset \mathcal{X}_{23}$, where $|\mathcal{I}_C^\star| = 2 \times 10^6$. For this set, we also got the same minimum convergence time ($T = 345$), like for $\mathcal{I}_C$, but of course, this still does not guarantee that it will be the case for the entire set $\mathcal{X}_{23}$.

Similar to the experiments made for a 5×5 grid, we computed the difference $\Delta T(\boldsymbol{x})$ for all configurations $\boldsymbol{x} \in \mathcal{I}_C^\star$ (Figure 8.5). We found

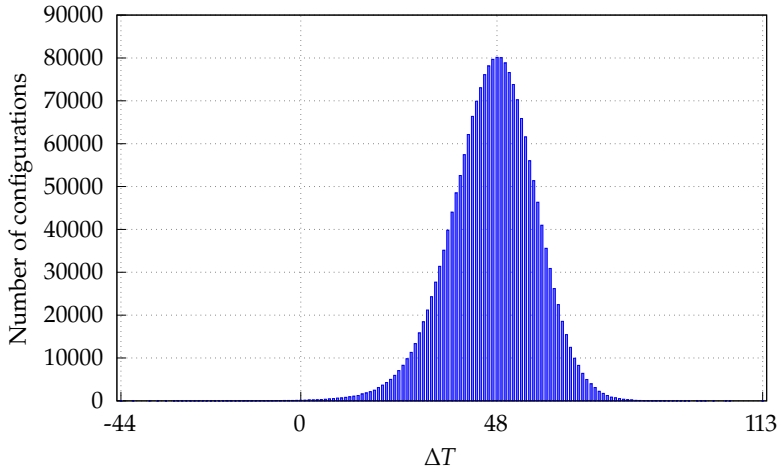
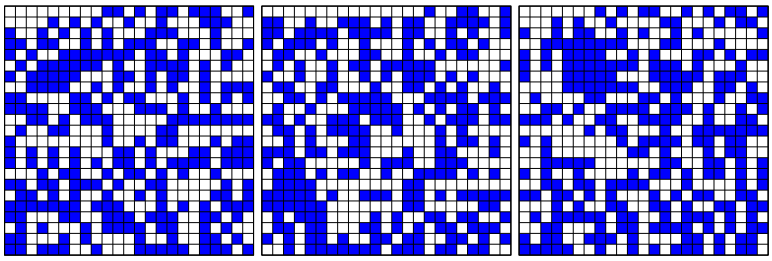


Figure 8.5: The number of configurations $\boldsymbol{x} \in \mathcal{I}_C$ with a given difference of convergence time $\Delta T(\boldsymbol{x})$ between rule $\mathbb{f}_d$ and $\mathbb{f}_\star$ for a 23×23 grid.

that there are 99.9% configurations that are solved up to 113 iterations faster by rule $\mathbb{f}_\star$. Some configurations for some given values of $\Delta T(\boldsymbol{x})$ are presented in Figure 8.6.



(a) $\Delta T(\boldsymbol{x}) = -44$ **(b)** $\Delta T(\boldsymbol{x}) = 0$ **(c)** $\Delta T(\boldsymbol{x}) = 113$

Figure 8.6: Configurations $\boldsymbol{x}$ for given values of $\Delta T(\boldsymbol{x})$ for a 23×23 grid.

Additionally, we checked which of the rotation-symmetric, density-conserving ACCAs is the fastest one. The best possible convergence time in this case is $T = 352$ corresponding to $\alpha = 0.249$. Thus, it is closer to $\frac{1}{4}$ than for the best rule for grid size $n = 5$. Moreover, for this experimental set-up, it is the only one rule with $T = 352$. The results of this experiment are presented in Figure 8.7. In this case the singularity mentioned before can be observed in the inset in Figure 8.7. Again, it appears that among rotation-symmetric ACCAs we can find rules that are better than the averaging rule and only slightly worse than the best rule. Hence, we conjecture that to find a local rule with the smallest convergence time, it is not necessary to search the entire 8-dimensional polytope P, but it suffices to search only through the one-dimensional subset of rotation-symmetric ACCAs.

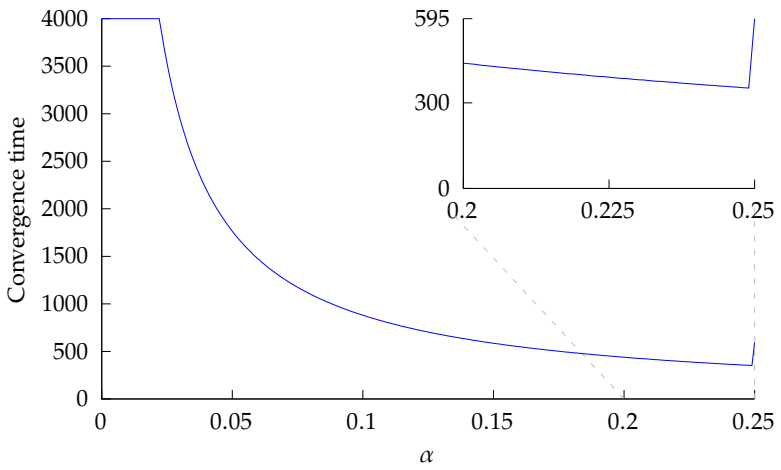


Figure 8.7: Results for rotation-symmetric density-conserving ACCA for a 23×23 grid.

Finally, we considered rules given by Eq. (3.19) and found that the behavior is similar to the one for $n = 5$. The minimal convergence time is $T = 739$, so it takes significantly longer than for the averaging rule. Also here, there are only three rules parameterized by

$$(b, d) \in \left\{\left(\frac{21}{100}, \frac{30}{100}\right), \left(\frac{29}{100}, \frac{29}{100}\right), \left(\frac{30}{100}, \frac{21}{100}\right)\right\}$$

that solve the relaxed DCP in the minimal convergence time $T = 739$. The results for this experiment are presented in Figure 8.8. Note that in

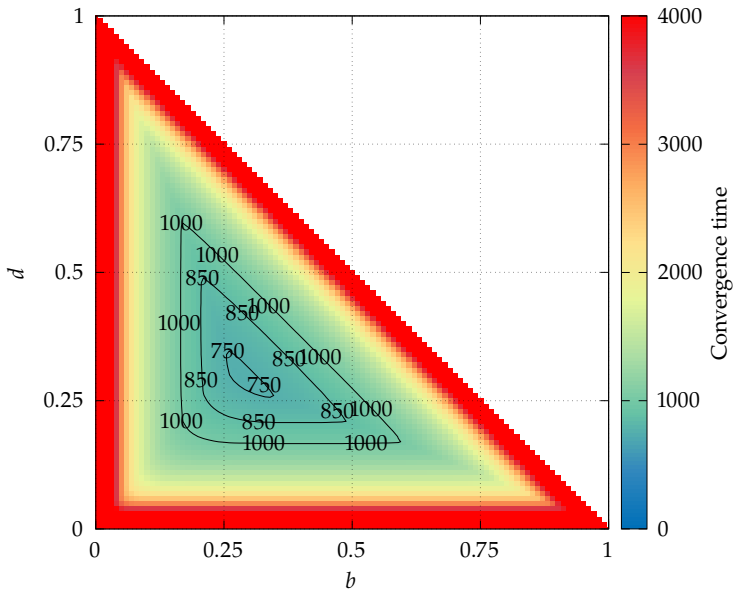


Figure 8.8: Results for density-conserving ACCA whose local rule is independent of y and v for a 23×23 grid.

this case the rules marked with red near the edges do not tell us that they do not solve the relaxed DCP, but that they require much more iterations than allowed in our experiment.

8.4 Conclusions

Although the computational complexity in the two-dimensional case is high, we expect that one does not need to examine the entire space of density-conserving rules described by an 8-dimensional polytope to get a

satisfactory solution, but that it suffices to examine the one-dimensional subspace of rotation-symmetric rules. For instance, for the two rule families discussed in Section 3.7, we were able to find a better rule among the rotation-symmetric (outer-totalistic) rules only. This result is important in view of similar tasks that are computationally even more challenging, like in settings with larger configurations.

Solving the Fixed-Length Two-Dimensional DCP

9

Let us suppose we have set up a machine with certain initial instruction tables, so constructed that these tables might on occasion, if good reason arose, modify those tables. One can imagine that after the machine had been operating for some time, the instructions would have altered out of all recognition, but nevertheless still be such that one would have to admit that the machine was still doing very worthwhile calculations. Possibly it might still be getting results of the type desired when the machine was first set up, but in a much more efficient manner. In such a case one would have to admit that the progress of the machine had not been foreseen when its original instructions were put in. It would be like a pupil who had learnt much from his master, but had added much more by his own work. (Alan Turing, 1947)

In this chapter, we consider the classical DCP in the context of two-dimensional ACCAs. Let us recall that we proved in Chapter 4 that there is no solution to the DCP. Later, in Chapter 6, we considered the problem in the case of a fixed number of cells in the one-dimensional case. As we saw, the analytical and computational complexity grows with the dimension, thus we will mostly focus on the experimental part of the research here, using the knowledge gained in previous chapters. We started this chapter by quoting Alan Turing's lecture at the London Mathematical Society in 1947, where he wanted to have a machine that would learn from experience. In our investigation, to help to find interesting solutions, we will use algorithms inspired by nature, where the *machine learns from experience*, starting from *certain initial instructions*.

9.1 Evolutionary Computation 105
9.2 History 106
9.3 DCP as an optimization problem 107
9.4 Differential Evolution . 109
9.4.1 Individuals 109
9.4.2 Population 109
9.4.3 Fitness function 110
9.4.4 Mutation 111
9.4.5 Crossover 111
9.4.6 Selection 112
9.5 Experiments for a 3×3 grid 112
9.5.1 Unique configurations . 112
9.5.2 Configurations with density $\rho \approx \frac{1}{2}$ 118
9.5.3 Rotation-symmetric rules 119
9.5.4 Observations 122
9.6 Experiments for a 5×3 grid 123
9.6.1 Rotation-symmetric rules 123
9.6.2 Basin of 3×3 rules . . . 125
9.6.3 Observations on smaller grids 126
9.7 Experiments for a 5×5 grid 127
9.8 Conclusions 128

9.1 Evolutionary Computation

The principles of biological evolution served as inspiration for *evolutionary computation* (EC). EC is a group of optimization algorithms used for finding optimal or near-optimal solutions to a problem. It can produce interesting results for a large variety of computationally hard problems.

The process presented in Figure 9.1 explains the main parts of EC. We start

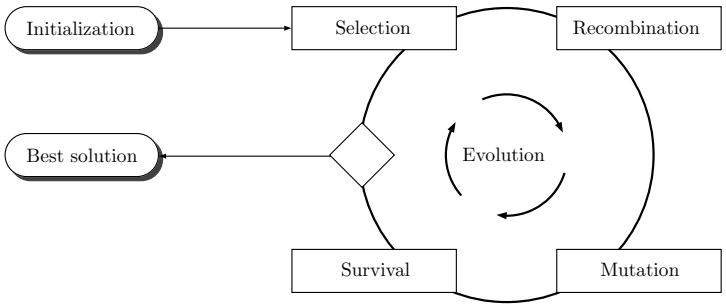


Figure 9.1: Diagram of an evolution process in evolutionary computation.

by defining an *individual* $\boldsymbol{\iota}$ describing a potential solution to a predefined

problem. This can be a vector or an array of values that contains a list of parameters describing the solution. We group these individuals into a set $\mathcal{P}$ that defines a *population* of size $|\mathcal{P}| = p$

$$\mathcal{P} = (\boldsymbol{\iota}_0, \ldots, \boldsymbol{\iota}_{p-1}) \, .$$

The population changes over time. By $\mathcal{P}^g$ we denote the population of the g-th *generation* of the EC. The first population, the *initial population* $\mathcal{P}^0$, can be generated randomly as a predefined set of individuals.

Each individual $\boldsymbol{\iota}$ is evaluated using a *fitness* function ϕ that evaluates how good it is at solving the problem. The higher the fitness value is, the better the individual solves the problem. This function tells us which individual or individuals are the fittest, *i.e.*, which ones in a generation solve the problem best. When we know what solution we want to obtain, we can measure the distance between the perfect solution and the solution represented by an individual. For that purpose, we use an *error function*, also known as a *cost function* or *loss function*. The closer the value of the cost function to 0, the better the result, where the value 0 indicates no difference from with the ideal solution.

The next generation comprises individuals from the previous generation modified by operators inspired by natural evolution. We start by *selecting* representative individuals from the last generation. Such individuals will be called *parents*. The next operator, *crossover*, is used during the *recombination* process and produces a temporary individual $\hat{\boldsymbol{\iota}}$ that comprises a combination of the characteristics of its parents. *Mutation* is another operator that might change some characteristic of the temporary individual $\hat{\boldsymbol{\iota}}$ by introducing random noise. We will refer to an individual that results from the above operations as an *offspring*. Based on natural selection, the next generation contains the same number of elements (this is not a fixed rule) as the initial population, but only with the fittest individuals. This is called a *survival* process, where only selected individuals move to the next generation. We repeat the entire process until we find the best individual with the highest fitness value or when we satisfy a termination condition.

9.2 History

We started this chapter by quoting Alan Turing's talk [77], where we can understand *instruction tables* as a set of commands in the modern programming language. A more comprehensive summary of the history of EC can be found in [78]. Here, we will only shortly present the history of EC with the focus on some main EC streams, their similarities and differences.

We can say that the era of evolutionary computation started with the first programmable digital computer ENIAC in 1945. In the mid -1950s, when first digital computers were already in use, the interest of digital models was rising. The first description of the use of an evolutionary process solving a problem on an IBM 704[1] computer was described in 1958 by Friedberg[2] [79].

1: The IBM 704 announced in 1954 was the first commercially available computer with floating point arithmetic commands. A division of a 36bit floating point numbers took 24ms.

2: Richard M. Friedberg (8 October 1935). An American theoretical physicist.

In the 1960s, three main optimization strategies inspired by nature were developed; two of them, almost at the same moment, in the USA.

The first one, *genetic algorithms* (GA), was developed by Holland[3] [80, 81]. In his work, the population contains individuals represented by binary strings. The population evolves using selection, mutation and recombination.

Another optimization strategy, *evolutionary programming* (EP), was developed by Fogel[4] [82]. EP was used to evolve finite-state machines with mutation as the main operator.

Evolution strategies (ES) are another type of optimization algorithms developed in the 1960s and improved in the 1970s by two students – Rechenberg[5] [83] and Schwefel[6] [84] – at the Technical University of Berlin in Germany. The problem that needed to be solved at that moment was shape optimization. Schwefel described the first analytical findings and computer experiments results from a Zuse Z23[7] in his diploma thesis. The algorithm only used a simple mutation-selection scheme. As with many optimization methods, the performance and results of ES also depend on few parameters. That resulted in an extension of ES where the population contains μ individuals and the size of the offspring set λ is defined:

- $(\mu + \lambda)$ type : $\lambda \geq 1$ offsprings create a trial population of size $(\mu + \lambda)$, where the worst λ individuals do not move to the next generation,
- (μ, λ) type : $\lambda > \mu$, *i.e.*, the selection is limited to the offspring, and the parents do not move to the next generation.

A more comprehensive description of ES can be found in [85] and [86].

3: John Henry Holland (2 February 1929 – 9 August 2015). An American scientist, professor of psychology, electrical engineering and computer science. Pioneer of genetic algorithms.

4: Lawrence Jerome Fogel (2 March 1928 – 18 February 2007). An American scientist. He was not only a pioneer of evolutionary programming. One of his inventions is active voice cancellation, used heavily in cockpits to reduce the noise.

5: Ingo Rechenberg (20 November 1934). A German computer scientist.

6: Hans Paul Schwefel (4 December 1940). A German computer scientist.

7: The Z23 computer was first delivered in 1961. A division of a 38 bit floating point number took 20ms.

9.3 The Density Classification Problem as an optimization problem

In this section, we will reformulate the classical DCP as an optimization problem, which enables us to use evolutionary search methods. We consider the classical DCP on a two-dimensional grid $\mathcal{C}_2$. Let us recall that in Section 2.1, we defined the two-dimensional grid $\mathcal{C}_2$ of a fixed size $n \times m$. The reformulation of Definition 4.2.1 to this case gives the following definition.

Definition 9.3.1 *Let $\mathbb{F}$ be the global rule of a two-dimensional ACCA. We say that this ACCA solves the classical DCP on a grid $\mathcal{C}_2$ if for any initial configuration $\boldsymbol{x} \in \{0,1\}^{\mathcal{C}_2}$, it holds that:*

$$\begin{aligned} \rho(\boldsymbol{x}) < \frac{1}{2} &\Rightarrow \lim_{t\to\infty} \mathbb{F}^t(\boldsymbol{x}) = 0^{\mathcal{C}_2}\,, \\ \rho(\boldsymbol{x}) > \frac{1}{2} &\Rightarrow \lim_{t\to\infty} \mathbb{F}^t(\boldsymbol{x}) = 1^{\mathcal{C}_2}\,. \end{aligned} \tag{9.1}$$

We do not only want to get a binary answer, *i.e.*, whether or not the rule $\mathbb{F}$ solves the DCP, but also to find out how close the solution is to the correct answer. For that purpose, we introduce an error function $\theta(\boldsymbol{x})$, which for

a configuration $\boldsymbol{x}$ returns a value from the interval $[0, 1]$ informing us how well the configuration is classified. We assume that the value equals zero when Eq. (9.1) is satisfied, *i.e.*, if there is no difference between the final configuration and the expected outcome. In other words, assuming that

$$\boldsymbol{x}^{\infty} = \lim_{t \to \infty} \mathbb{F}^t(\boldsymbol{x}) \,, \tag{9.2}$$

we can rewrite Eq. (9.1) as

$$\theta(\boldsymbol{x}) = \begin{cases} 0 & \text{if } \rho(\boldsymbol{x}) < \frac{1}{2} \text{ and } \boldsymbol{x}^{\infty} = 0^{C_2} \,, \\ 0 & \text{if } \rho(\boldsymbol{x}) > \frac{1}{2} \text{ and } \boldsymbol{x}^{\infty} = 1^{C_2} \,, \\ 1 & \text{otherwise} \,. \end{cases} \tag{9.3}$$

We see in Eq. (9.3) that the value $\theta(\boldsymbol{x})$ gives a binary answer whether the configuration is classified or not. An alternative error function can be defined as

$$\theta(\boldsymbol{x}) = \begin{cases} \rho(\boldsymbol{x}^{\infty}) & \text{if } \rho(\boldsymbol{x}) < \frac{1}{2} \,, \\ 1 - \rho(\boldsymbol{x}^{\infty}) & \text{if } \rho(\boldsymbol{x}) > \frac{1}{2} \,. \end{cases}$$

This function returns the difference between the density of the final configuration $\boldsymbol{x}^{\infty}$ and the expected density. The third error function we suggest is

$$\theta(\boldsymbol{x}) = \begin{cases} \max\limits_{(i,j) \in C_2} x_{i,j}^{\infty} & \text{if } \rho(\boldsymbol{x}) < \frac{1}{2} \,, \\ 1 - \min\limits_{(i,j) \in C_2} x_{i,j}^{\infty} & \text{if } \rho(\boldsymbol{x}) > \frac{1}{2} \,, \end{cases} \tag{9.4}$$

where it is just the greatest difference between any cell value of the final and the expected configurations.

Having the error function for a single configuration $\boldsymbol{x}$, we introduce an error function for the set of configurations

$$\Theta(\mathcal{I}) = \frac{1}{|\mathcal{I}|} \sum_{\boldsymbol{x} \in \mathcal{I}} \theta(\boldsymbol{x}) \,.$$

This function returns the average error of all configurations $\boldsymbol{x} \in \mathcal{I}$. As we restricted the function $\theta(\boldsymbol{x})$ to $[0, 1]$, this ensures that also $\Theta(\mathcal{I}) \in [0, 1]$. The optimal value of $\Theta(\mathcal{I})$ is zero, so the global rule $\mathbb{F}$ classifies all $\boldsymbol{x} \in \mathcal{I}$. Apart from the error function, we can also use a fitness function

$$\Phi(\mathcal{I}) = 1 - \Theta(\mathcal{I}) \,, \tag{9.5}$$

where the optimal value is one in this case.

In our further considerations, we use Eq. (9.4) as the error function θ. We note that for *in silico* experiments we have to deal with the size of the configuration set and also with the response time of the experiments, especially when evaluating Eq. (9.2). In addition, we have to deal with the computational imperfection in floating point arithmetic and find a tradeoff between the speed and precision of our simulations.

9.4 Differential Evolution

In this section, we will recall *differential evolution* (DE), one of the EC optimization methods used in our experiments. We will only focus on the classical version of the algorithm [87] and explain it in the context of ACCAs and the DCP as an optimization problem, as presented in Section 9.3. As described in Section 9.1, we need to define the individuals (their representation), the population structure, the fitness function and the procedure of reproduction: selection, crossover and mutation.

9.4.1 Individuals

When we recall Definition 4.2.1, we know that our goal is to find a local rule such that for any initial configuration $\boldsymbol{x}$, applying the global rule $\mathbb{F}$ leads to a homogeneous configuration of all zeros or ones depending on the initial configuration density.

A potential solution for the classical DCP in DE is represented as an individual $\boldsymbol{\iota}$ that is a vector of parameters. The individual can be described by the full LUT of a two-dimensional ACCA (see Table 2.2)

$$(l_0, l_1, l_2, \ldots, l_{29}, l_{30}, l_{31}) \ ,$$

or only by s parameters that can be expanded to the full LUT, thus, we will represent the individual as

$$\boldsymbol{\iota} = (\iota_0, \ldots, \iota_{s-1}).$$

9.4.2 Population

By a population we mean a set $\mathcal{P}$ of p individuals

$$\mathcal{P} = \{\boldsymbol{\iota}_0, \ldots, \boldsymbol{\iota}_{p-1}\},$$

which is represented in Figure 9.2. We will denote the population of

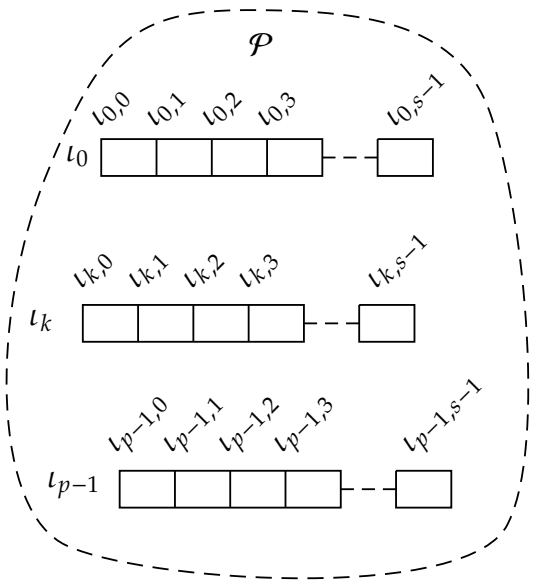


Figure 9.2: Graphical representation of a population $\mathcal{P}$ of individuals $\boldsymbol{\iota}_0, \ldots, \boldsymbol{\iota}_{p-1}$.

the g-th generation of DE by $\mathcal{P}^g$. Here, $g \in \{0, \ldots, G\}$ is the generation number and G is the maximal number of generations allowed. By $\boldsymbol{\iota}_k^g$ we will denote the k-th individual in the population $\mathcal{P}^g$. Moreover we will refer with $\iota_{k,i}^g$ to the i-th entry of the individual

$$\boldsymbol{\iota}_k^g = \left(\iota_{k,0}^g, \ldots, \iota_{k,s-1}^g\right).$$

The initial population $\mathcal{P}^0$ is generated by setting each $\iota_{k,i}^0$ of each individual $\boldsymbol{\iota}$ with a uniformly distributed random number within the range $[0, 1]$.

9.4.3 Fitness function

To measure how well an individual solves our problem, we will measure its fitness on a set of configurations $\mathcal{I}$. Let us recall that in Section 9.3 we defined DCP as an optimization problem, where the error function in Eq. (9.4) evaluates the configuration for an infinite amount of time. This case is not valid anymore for *in silico* experiments. For that purpose, we introduce an error function θ_ι, which after applying the global rule $\mathbb{F}_\iota$ corresponding to the individual $\boldsymbol{\iota}$ T times, measures the distance between the final configuration and the configuration of all 0s or all 1s, depending on the initial configuration's density. We can rephrase Eq. (9.4) to

$$\theta_\iota(\boldsymbol{x}) = \begin{cases} \max\left(\mathbb{F}_\iota^T(\boldsymbol{x})\right) & \text{, if } \rho(\boldsymbol{x}) < \frac{1}{2}\,, \\ 1 - \min\left(\mathbb{F}_\iota^T(\boldsymbol{x})\right) & \text{, if } \rho(\boldsymbol{x}) > \frac{1}{2}\,, \end{cases} \tag{9.6}$$

so $\theta_\iota(\boldsymbol{x})$ is the greatest difference of a cell value of the final configuration after the T-th iteration of $\mathbb{F}_\iota(\boldsymbol{x})$ compared with 0, when the initial density is below $\frac{1}{2}$, or with 1, in the other case. It is obvious that the values of error function are limited to the interval $[0, 1]$. Having defined the error function, we redefine the *fitness function* from Eq. (9.5) in the following way

$$\Phi_{\mathcal{I}}(\boldsymbol{\iota}) = 1 - \frac{1}{|\mathcal{I}|} \sum_{\boldsymbol{x} \in \mathcal{I}} \theta_\iota(\boldsymbol{x})\,.$$

As for the error function, the fitness function is limited only to values in $[0, 1]$. After limiting the evaluation of a configuration $\boldsymbol{x}$ to only T steps, we have to note that it is hardly possible to reach the perfect fitness value of 1. Here, the higher the value, the better, *i.e.*, the closer to 1, the better (fitter) the individual $\boldsymbol{\iota}$ is. Still, after limiting the evaluation of configuration $\boldsymbol{x}$, the fitness function is time and resource-demanding. It requires performing multiple long running operations, since each configuration needs to be evaluated T times.

For vast grids, as the number of initial configurations grows exponentially, it might be even impossible to evaluate the fitness function in a reasonable amount of time. To overcome this hurdle, we will run the fitness function in parallel for each configuration first and aggregate the results at the end. Also for larger sets, we will divide the set of initial configurations and define a *training set* $\mathcal{I}_T$ and a *validation set* $\mathcal{I}_V$. The training set $\mathcal{I}_T$ will be used in the standard DE process, whereas the validation set $\mathcal{I}_V$ will be used only after a certain number of generations or at the end of the

DE process to analyze the behavior of the individual on a broader set of configurations, even with a higher T.

9.4.4 Mutation

Having a population $\mathcal{P}$, we can produce a population $\hat{\mathcal{P}}$ by using *differential mutation*. The process creates a k-th individual $\hat{\boldsymbol{\iota}}$ by adding a randomly sampled difference of two individuals b and c to a third individual a

$$\hat{\boldsymbol{\iota}}_k^g = \min\left(1, \max\left(0, \boldsymbol{\iota}_a^g + \mu\left(\boldsymbol{\iota}_b^g - \boldsymbol{\iota}_c^g\right)\right)\right) ,$$

where the indices a, b, c, k are pairwise different. We also ensure that the result of mutation does not go beyond the interval $[0, 1]$. To control the evolution rate, a scale factor $\mu \in (0, 1)$ is used. In our experiments, we will use $\mu = \frac{1}{2}$. After this process, we have a trial population of p mutant individuals $\hat{\iota}$. The process is visualized in Figure 9.3.

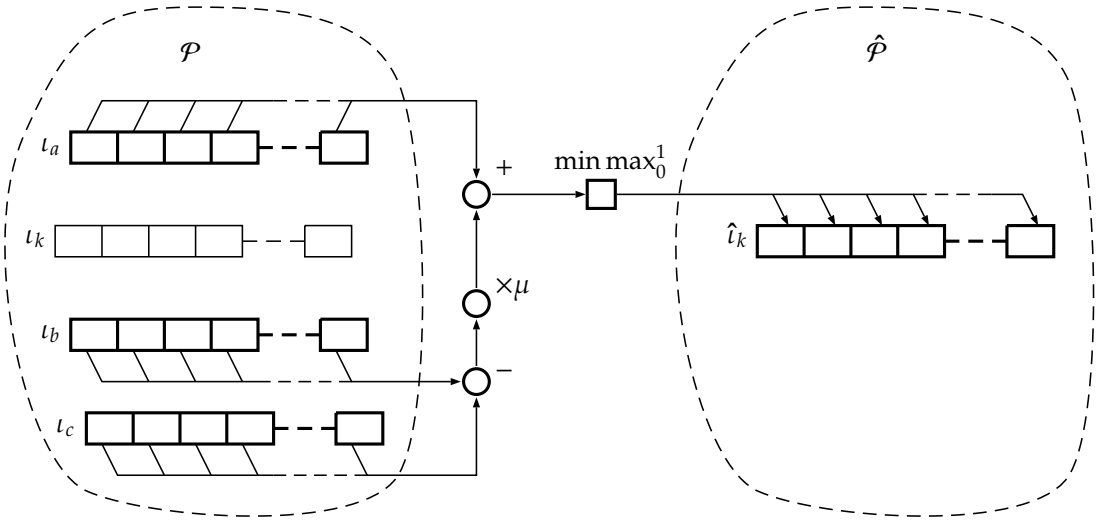


Figure 9.3: Graphical representation of a differential mutation.

9.4.5 Crossover

Crossover, also referred as a *discrete recombination*, is the next operation used in DE. Let $\xi(i)$ be a function that returns a random number following the uniform distribution over the interval $[0, 1]$. Let ξ_k be a random number following the discrete uniform distribution over the set $0, \ldots, p-1$, in other words, a variable that holds a random value for each individual in a generation. In the crossover operation, each entry $\hat{\boldsymbol{\iota}}_{k,i}$ of the k-th individual $\boldsymbol{\iota}$ is replaced with the parent entry $\boldsymbol{\iota}_{k,i}$ when the following criterion is met

$$\hat{\boldsymbol{\iota}}_k = \left(\begin{cases} \hat{\boldsymbol{\iota}}_{k,i} & \text{, if } \xi(i) \leq \xi \text{ or } i = \xi_k \\ \boldsymbol{\iota}_{k,i} & \text{, otherwise} \end{cases}\right)_{i=0}^{n} ,$$

where $\xi \in [0, 1]$ is a crossover probability that controls the fraction of mutant vector parameters that are copied to the individual. In our experiments, we will use $\xi = 0.9$.

9.4.6 Selection

After the crossover, we invoke a selection. In this process, the individual $\hat{\boldsymbol{\iota}}$ replaces $\boldsymbol{\iota}$ in the next generation if

$$\boldsymbol{\iota}_k^{g+1} = \begin{cases} \hat{\boldsymbol{\iota}}_k^g & \text{, if } \Phi_I(\hat{\boldsymbol{\iota}}_k^g) \geq \Phi_I(\boldsymbol{\iota}_k^g) \text{ ,} \\ \boldsymbol{\iota}_k^g & \text{, otherwise.} \end{cases}$$

In other words, the individual is replaced by the offspring when the offspring gives the same or a better result. Once the selection is invoked for all individuals, the population $\hat{\mathcal{P}}$ replaces $\mathcal{P}$ and creates a new generation. This process is repeated until the optimum (best individual) is found or a termination criterion is met - in our case, generation G is reached.

9.5 Experiments for a 3×3 grid

We start our experiments with the smallest possible square grid 3×3, for which the set of all binary configurations $\mathcal{I}_C$ contains only 512 configurations. This easiest setup allows us to analyze the behavior of DE runs quite fast. We divide the experiment into three subcategories.

9.5.1 Unique configurations

In this experiment, we reduce the set of binary configurations for the 3×3 grid to unique configurations only, *i.e.*, those whose shifts are equivalent to a representative as explained in Section 2.3. As a result, the set $\mathcal{I}_C$ is reduced to 64 unique configurations, which reduces the required time spent on the evaluation of the fitness function by a factor eight.

The training set and the validation set are identical $\mathcal{I}_T \equiv \mathcal{I}_V \equiv \mathcal{I}_C$ and contain the same configurations. We use the following parameters in the DE setup: the termination condition for the number of generations $G = 200$, the termination condition for the number of configuration iterations $T = 30$, the validation is invoked on every 10-th generation. During validation we modify only the termination condition for the number of configuration iterations to $2T$. As we are looking for ACCA rules which at least need to be double-legal, the population $\mathcal{P}$ is made up of individuals $\boldsymbol{\iota}$ that represent 30 LUT coefficients, excluding only the boundary neighborhoods l_0, l_{31}

$$\boldsymbol{\iota} = (l_1, l_2, \ldots, l_{29}, l_{30}) \text{ .}$$

The initial population $\mathcal{P}^0$ is initialized by assigning each coefficient $\iota_{0,i}$ of the i-th individual a random number in $[0, 1]$. The size of $|\mathcal{P}| = 30$ is equal to the dimension of the individual. We ran the DE algorithm 1000 times, resulting in 1000 different initial populations.

Figure 9.4 shows the distribution of the maximum individual fitness in a generation for all 1000 experiments. From each DE run, we picked the

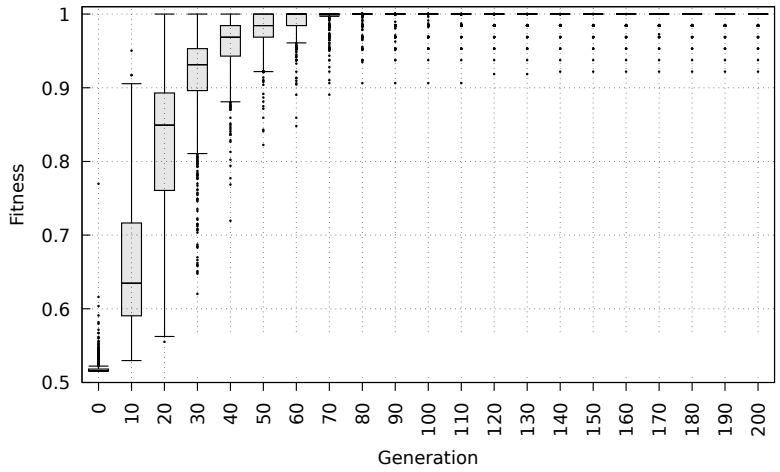


Figure 9.4: Distribution of individual fitness for 1000 DE runs for the problem of finding an ACCA that solve the DCP using all binary configurations in the training set on the 3×3 grid.

fittest individual from the validation runs only. As a result, we found 959 unique ACCAs with $\Phi_I(\iota) = 1$, *i.e.*, ACCAs that solve the DCP in the classical sense on the 3×3 grid for all binary configurations. Only 41 runs resulted in a situation where the best individual's fitness was $\Phi_I(\iota) < 1$.

First, we present and describe a DE run, where an individual representing an ACCA that solves the classical DCP described by the LUT in Table 9.1 was found. The fitness value of the best individual in a generation is

Table 9.1: LUT of a two-dimensional ACCA rule that solves the classical DCP for the 3×3 grid found by the DE algorithm.

l_0	0	l_1	0.0226502054303856	l_2	0.0514413448236731	l_3	0.2963377655833841
l_4	0	l_5	0.0417211954604394	l_6	0.3242656319265483	l_7	0.9916028466046108
l_8	0	l_9	0	l_{10}	0.0784318040417061	l_{11}	0.2235551571809470
l_{12}	0.1325716987769570	l_{13}	0.7571973602095508	l_{14}	0.7172918374457891	l_{15}	0.9672432880422694
l_{16}	0.0638122748935461	l_{17}	0.4788203074797027	l_{18}	0.9512678576790132	l_{19}	0.7657430254189000
l_{20}	0	l_{21}	0.8330410602671330	l_{22}	0.4577090530313767	l_{23}	0.9840883754363199
l_{24}	0.0724604376522259	l_{25}	0.9404407932523569	l_{26}	1	l_{27}	0.9948595072812375
l_{28}	0.8554254171554112	l_{29}	1	l_{30}	0.9864446823866924	l_{31}	1

presented in Figure 9.5. We see that the value of the best individual

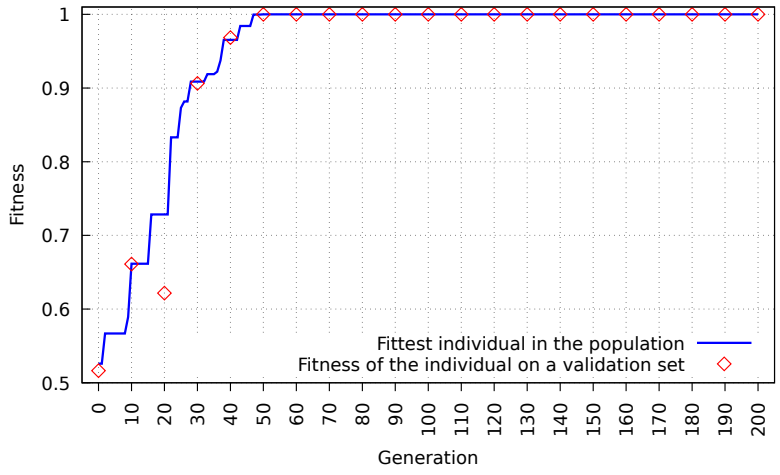
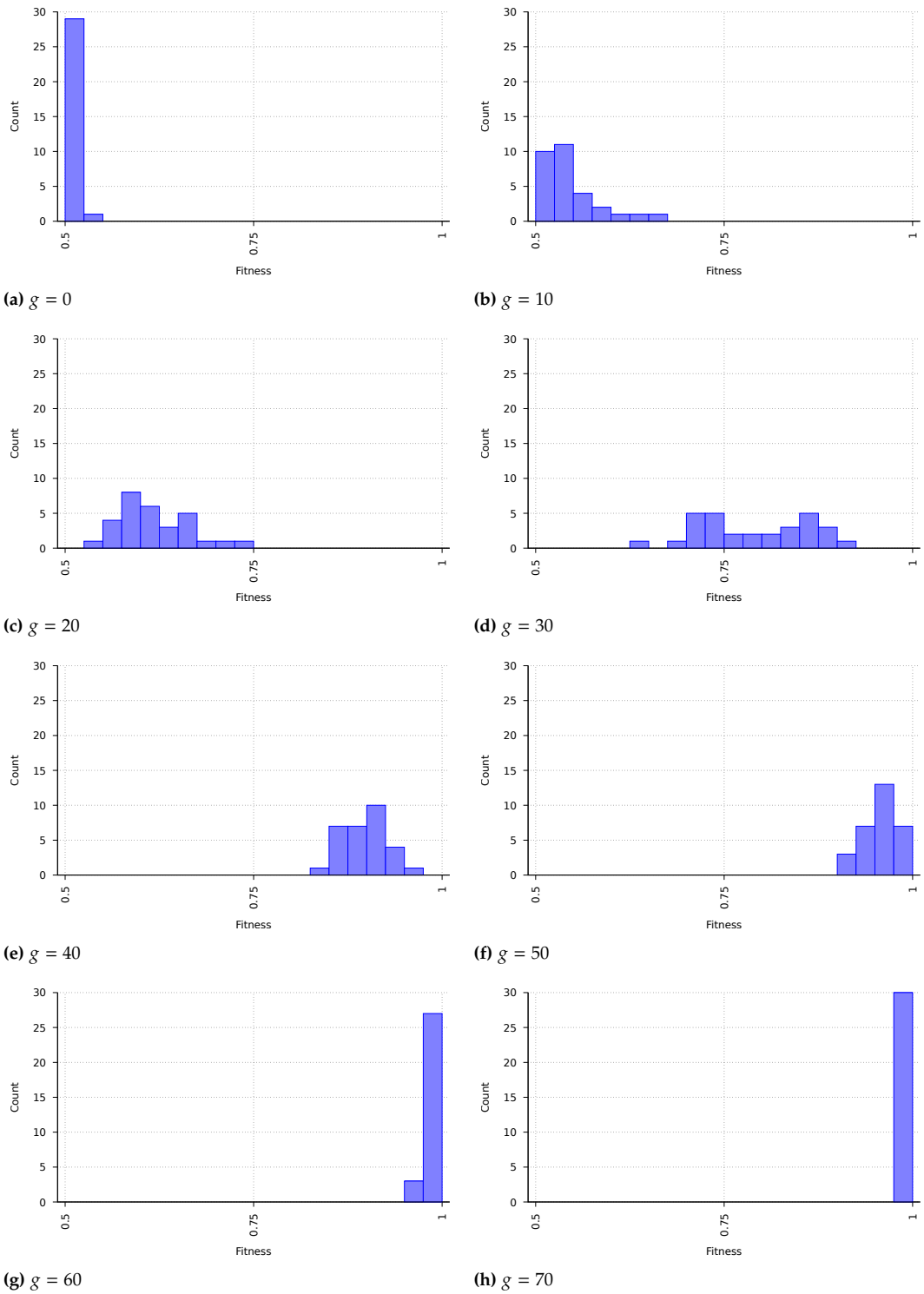


Figure 9.5: Fitness value of the best individual in a generation with marked fitness of the best individual in the validation process for an individual representing an ACCA solving the DCP on the 3×3 grid.

rises to 1 near the 50-th generation. Also, we can see that the validation

fitness value of the best individual found in the validation process for a generation also rises to 1, but is not correlated with the fitness value from the training set. The aberrance can be especially seen in generations 0 and 20. This is related to the definition of our fitness function, where for a configuration x^t after t evaluations, we take the maximal difference from the desired value.



(a) $g = 0$ **(b)** $g = 10$

(c) $g = 20$ **(d)** $g = 30$

(e) $g = 40$ **(f)** $g = 50$

(g) $g = 60$ **(h)** $g = 70$

Figure 9.6: Fitness histograms of individuals from the population in selected generations of the DE run for the problem of finding an ACCA that solves the classical DCP for the 3×3 grid.

The evolution of the fitness values of all individuals in the validation generation can be observed in the histograms in Figure 9.6. The initial population contains individuals with a fitness around $\frac{1}{2}$ with one outlier that is better (Figure 9.6a). The histograms in Figures 9.6b–9.6g present the evolution of the fitness in the population. In all generations better individuals with higher fitness values, pull other individuals to a better solution. In generation 70, presented in Figure 9.6h, we see all individuals to have a fitness value near the highest potential value. This process shows that the overall fitness of the population improves with each generation. In our further experiments, we will merge the fitness diagram of a single experiment in Figure 9.5 and the fitness histogram from Figure 9.6 into one Figure 9.7. In this figure, we see both the fitness of the best individual in a generation and the fitness value distribution in the population in the validation generation.

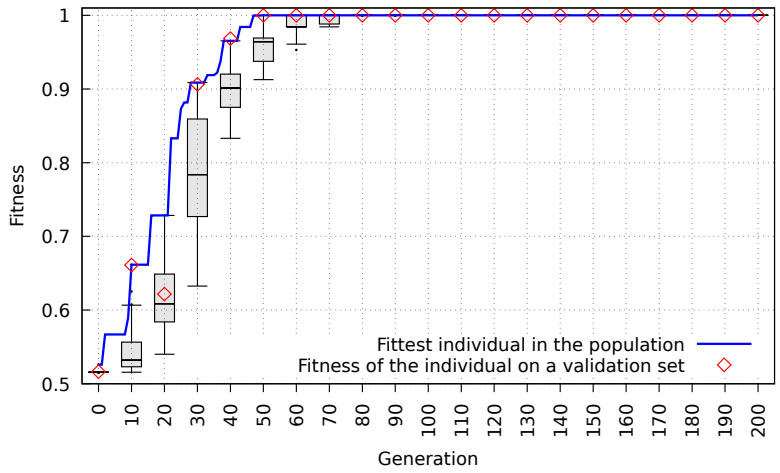
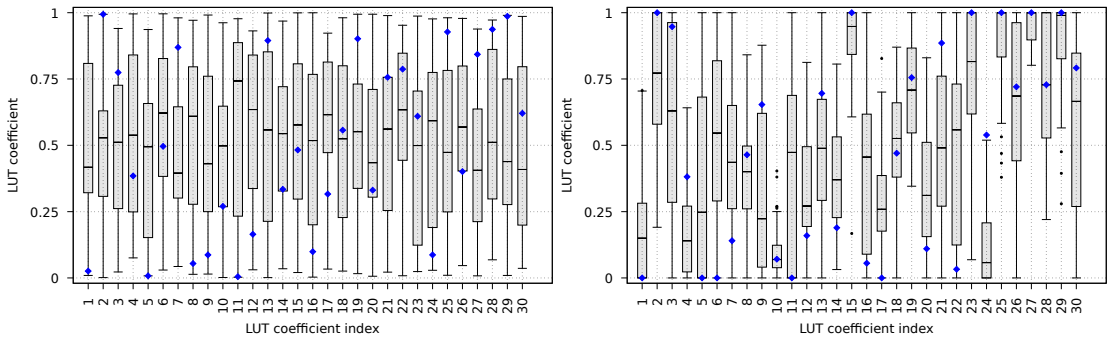
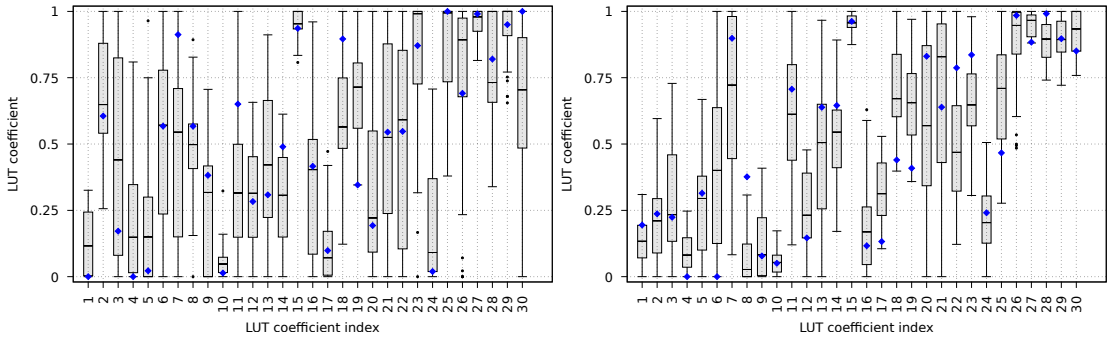


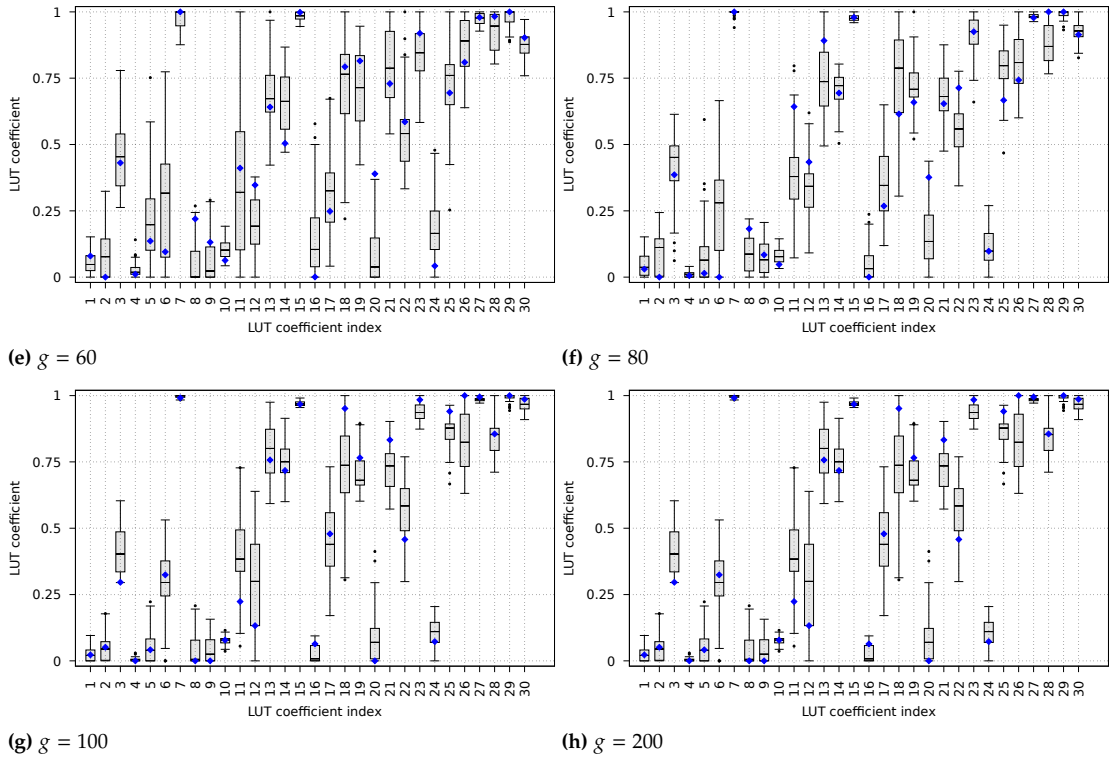
Figure 9.7: Fitness value of the best individual in a generation with marked fitness of the best individual in the validation process for an individual representing an ACCA solving the DCP on the 3×3 grid.



(a) $g = 0$ **(b)** $g = 10$



(c) $g = 20$ **(d)** $g = 40$



(e) $g = 60$ **(f)** $g = 80$ **(g)** $g = 100$ **(h)** $g = 200$

Figure 9.8: LUT coefficient distribution in selected generations for a DE run for the problem of finding an ACCA that solves the classical DCP for the 3×3 grid. Blue rotated squares represent the coefficient values of the best individual in the generation.

In Figure 9.8, we observe how the values of the individual (the LUT coefficients) change over generations. Figure 9.8a shows that the LUT coefficients are indeed almost uniformly distributed across the initial population. Over the next generations (Figures 9.8b–9.8f) the values converge to a shorter interval. The population stabilizes around the 100-th generation (Figure 9.8g), where we do not observe any improvement and the distribution of the LUT coefficients values no longer changes.

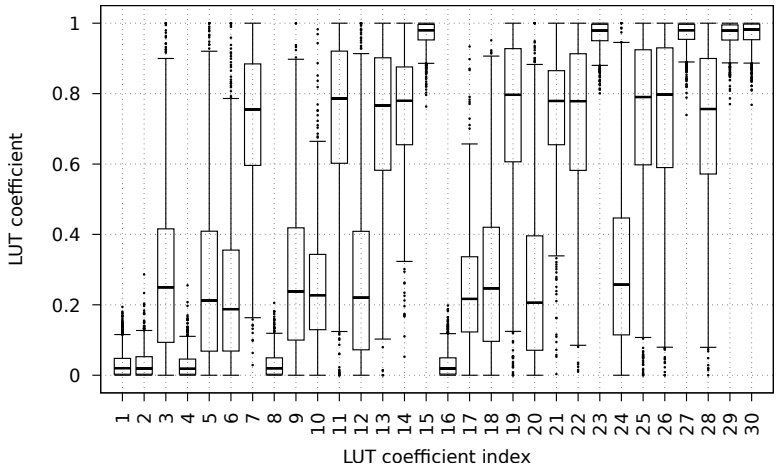


Figure 9.9: LUT coefficient distribution of the 959 best ACCAs found in DE experiments for the problem of finding an ACCA that solves the classical DCP for the 3×3 grid.

Figure 9.9 shows the LUT coefficient distribution of all the 959 individuals where the fitness value was best, *i.e.*, $\Phi_I(\iota) = 1$. We can see that the interval of the values for each coefficient becomes more narrow. We may also observe that few LUT entries like $l_1, l_2, l_4, l_8, l_{16}$ and $l_{15}, l_{23}, l_{27}, l_{29}, l_{30}$ have a similar distribution of values. Recalling Eq. (3.21), the above

observation reflects the behavior of rotation-symmetric rules.

Let us present one of the 41 DE runs where $\Phi_I(\iota) < 1$. An example of a LUT is presented in Table 9.2. Figure 9.10 shows fitness values of the best

Table 9.2: An example of a LUT of a two-dimensional ACCA rule found by the DE algorithm that do not solve the classical DCP for all configurations on the 3×3 grid.

l_0	0	l_1	0.0694483294696466	l_2	0.0205768632252970	l_3	0.3936142824964936
l_4	0.0324256239840529	l_5	0.6413351118370271	l_6	0.3420886711599403	l_7	1
l_8	0.0390656643647346	l_9	0.3323542788090380	l_{10}	0.5145222195645875	l_{11}	0.9337041925574603
l_{12}	0.4083933032081399	l_{13}	0.5558771701745073	l_{14}	0.3256708833263801	l_{15}	0.9996975188577638
l_{16}	0.0061853565507727	l_{17}	0.3477703425016097	l_{18}	0.0318818957581097	l_{19}	0.9516210370447654
l_{20}	0.0439863261339373	l_{21}	0.0098972290556301	l_{22}	0.8140427796400955	l_{23}	0.9974848794610039
l_{24}	0.0058144746957105	l_{25}	0.9227795261776912	l_{26}	0.9402703718388361	l_{27}	0.9353010574907661
l_{28}	0.2382834336453912	l_{29}	0.9878185427694743	l_{30}	0.9307289703828943	l_{31}	1

individual in a generation and the distribution of all the fitness values of the individuals in the population during validation process. We see that

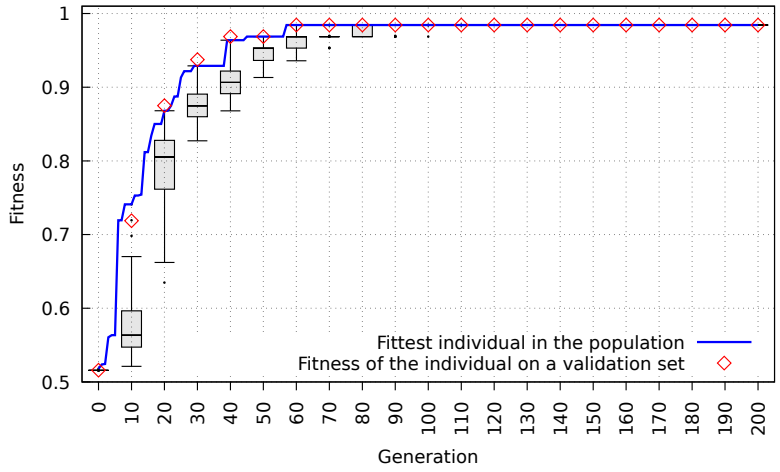
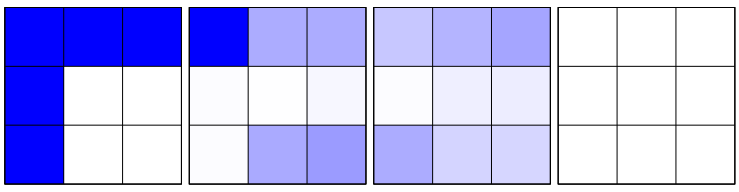


Figure 9.10: Fitness value of the best individual in a generation with marked fitness of the best individual in the validation process for an individual representing an ACCA that does not completely solve the classical DCP on the 3×3 grid.

here the maximal fitness reaches only 0.984. For the LUT presented in Table 9.2 only one configuration is wrongly classified. Figure 9.11 shows a sample evolution of an ACCA rule defined by the above LUT.



(a) x **(b)** $\mathbb{F}^1(x)$ **(c)** $\mathbb{F}^2(x)$ **(d)** $\mathbb{F}^9(x)$

Figure 9.11: Configuration x and a sample evolution of an ACCA rule found by the DE algorithm that does not solve the classical DCP on the 3×3 grid.

In this experiment, we used the easiest possible approach, where we just tested all the initial configurations for the 3×3 grid. This will not be the case for other grids where more initial configurations are available, thus we will try other approaches. The first one is used by many researchers and uses only configurations that have a density close to $\frac{1}{2}$. The idea behind the next experiment is a result of our observation of the rotation-symmetric rules.

9.5.2 Configurations with density $\rho \approx \frac{1}{2}$

This experiment has almost the same setup as in the previous section. The only difference is that the training set $\mathcal{I}_T$ is reduced to only configurations with a density close to $\frac{1}{2}$, *i.e.*, configurations with only four or five cells equal to 1. As a result, $|\mathcal{I}_T| = 28$ and $\mathcal{I}_V \equiv \mathcal{I}_C$, where $|\mathcal{I}_C| = 64$. Figure 9.12 shows the distribution of the best individual in a generation for all 1000 runs. We see no significant differences when comparing the distribution

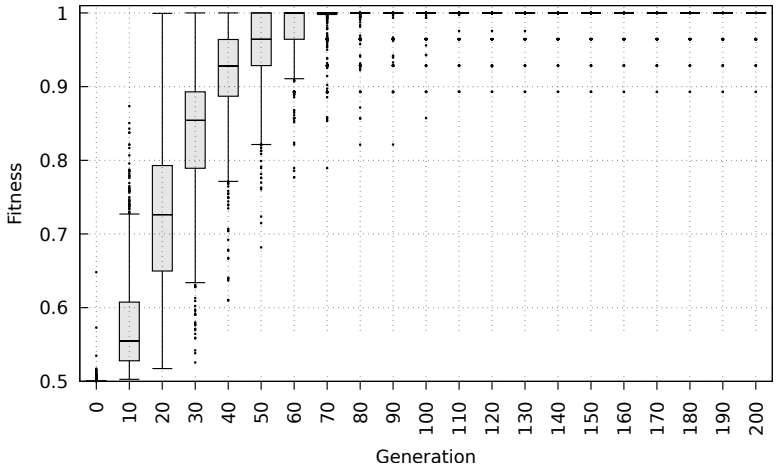


Figure 9.12: Distribution of individual fitness for 1000 DE runs for the problem of finding an ACCA rule that solves the DCP using only binary configurations with a density close to $\frac{1}{2}$ in the training set on the 3×3 grid.

with that of the previous experiment presented in Figure 9.4. Also here, the fitness values close to 1 are in the 70-th generation. Using this setup, 1000 DE runs found 957 rules that solve the DCP for the 3×3 grid for all binary configurations, thus the ratio is the same when compared with the previous experiment setup.

The LUT of one of the 957 rules is presented in Table 9.3. The fitness

Table 9.3: LUT of a two-dimensional ACCA rule that solves the classical DCP for the 3×3 grid found by the DE algorithm where only configurations with a density close to $\frac{1}{2}$ were used during the evolution.

l_0	0	l_1	0.0094220788056185	l_2	0.0089778109738877	l_3	0.0506542601172838
l_4	0.0148475414685278	l_5	0.3582985356212308	l_6	0	l_7	0.4368476624947731
l_8	0.0349355409111639	l_9	0.1542649789427312	l_{10}	0.0564312532604916	l_{11}	0.7700410565079709
l_{12}	0.3954581037799654	l_{13}	0.4888848643107001	l_{14}	0.9479710256801075	l_{15}	0.9995138259991005
l_{16}	0.0413349007990867	l_{17}	0.5206026658066896	l_{18}	0	l_{19}	0.8629755604608232
l_{20}	0.5770411756999720	l_{21}	0.9287420382591846	l_{22}	0.8481530205452407	l_{23}	0.9973319317629893
l_{24}	0.0108969658195211	l_{25}	0.7495690598626537	l_{26}	0.9811385956510490	l_{27}	0.9915991262973569
l_{28}	0.9485404068339514	l_{29}	1	l_{30}	0.7718279184631588	l_{31}	1

evolution and the fitness distribution of individuals for the LUT are presented in Figure 9.13. Comparing the evolution with the one presented in Figure 9.10, we see almost no differences, except that in many places the validation fitness is better for the best individual than the training fitness. This observation can be explained by the fact that the validation fitness uses also configurations with lower (or higher) densities than the ones used during the evolution process and as a result those single configurations evolve to the desired value faster, yielding a better average fitness value.

In Table 9.4 we present the LUT of one of the 43 rules that does not solve the problem for all configurations. Figure 9.14 illustrates the fitness value

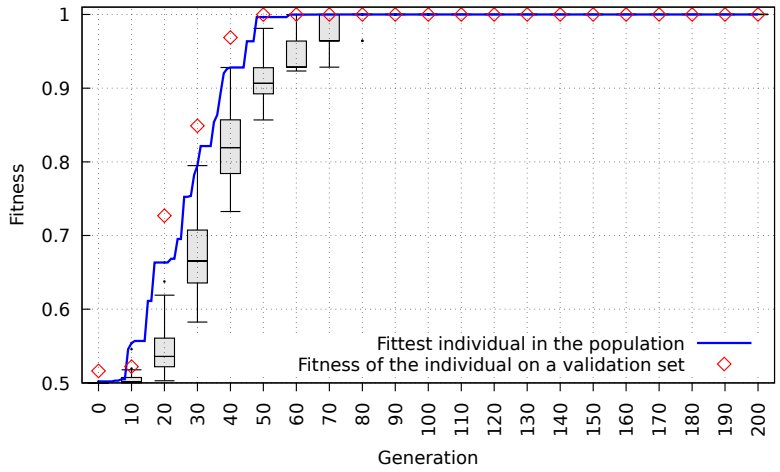


Figure 9.13: Fitness value of the best individual in a generation with marked fitness of the best individual in the validation process for an individual representing an ACCA rule solving the classical DCP on the 3×3 grid with only configurations with the density close to $\frac{1}{2}$ used during evolution.

Table 9.4: An example of a LUT of a two-dimensional ACCA rule found by DE algorithm that does not solve the classical DCP for all configurations on the 3×3 grid, but solves all configurations with the density $\rho \approx \frac{1}{2}$.

l_0	0	l_1	0	l_2	0.0047653400420446	l_3	0.0371745864049720
l_4	0.0228212872605086	l_5	0.8106790297235034	l_6	1	l_7	0.9976111548245561
l_8	0.0811382991601932	l_9	0.1769788136022301	l_{10}	0	l_{11}	0
l_{12}	0	l_{13}	1	l_{14}	0.9757222671851498	l_{15}	0.9171309982216329
l_{16}	0.0389240965820177	l_{17}	0	l_{18}	0.1229722494439303	l_{19}	0
l_{20}	0.0514454302416031	l_{21}	0.8472645520181108	l_{22}	1	l_{23}	1
l_{24}	0.6006751238881092	l_{25}	0.5846712066618136	l_{26}	1	l_{27}	1
l_{28}	0.7843282532058298	l_{29}	1	l_{30}	1	l_{31}	1

with a density close to $\frac{1}{2}$ of the best individual in a generation for that LUT. The most interesting observation is that in a certain generation, the

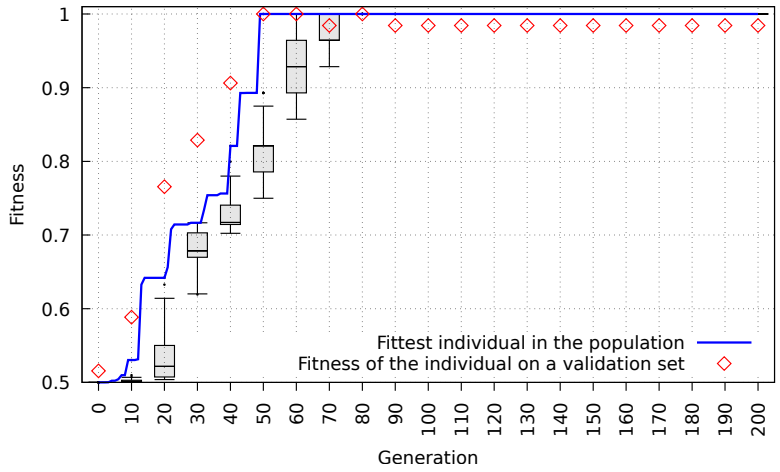
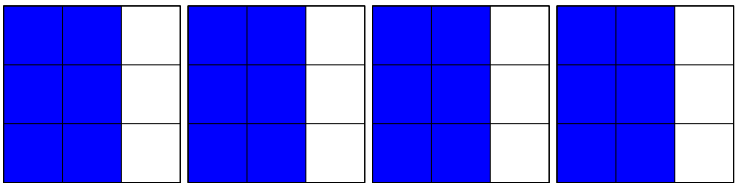


Figure 9.14: Fitness value of the best individual in a generation solving the classical DCP on the 3×3 grid. Only configurations with the density close to $\frac{1}{2}$ were used in the learning set.

fitness value reaches the best value, while the validation fitness does not reach it. Figure 9.15 shows a configuration evaluation that was not classified by the rule. We see that the configuration does not evolve to a homogeneous configuration.

9.5.3 Rotation-symmetric rules

In the experiments presented in Chapter 8, we showed that rotation-symmetric rules also give excellent solutions for solving the relaxed



(a) $t = 0$ **(b)** $t = 1$ **(c)** $t = 2$ **(d)** $t = 9$

Figure 9.15: Configuration $\boldsymbol{x}$ and sample iterations of an ACCA rule found by the DE algorithm using only configurations with a density close to $\rho \approx \frac{1}{2}$ for the 3×3 grid that does not evolve to the correct homogeneous configuration.

DCP for fixed-length configurations. In this section, we will run an experiment where only rotation-symmetric rules will be used in the DE runs to find ACCAs that solve the classical DCP for a 3×3 grid. Recalling Definition 3.7.1 and Theorem 3.7.1, the definition of an individual for rotation-symmetric rules, excluding also boundary neighborhoods, is reduced to only 10 LUT coefficients

$$\boldsymbol{\iota} = (l_1, l_3, l_4, l_5, l_7, l_{10}, l_{,11}, l_{14}, l_{15}, l_{27}) \,. \tag{9.7}$$

We start the experiment with the same setup as before. The training set $\mathcal{I}_T$ contains only configurations with the density close to $\frac{1}{2}$, resulting in $|\mathcal{I}_T| = 28$ and $\mathcal{I}_V \equiv \mathcal{I}_C$, where $|\mathcal{I}_C| = 64$.

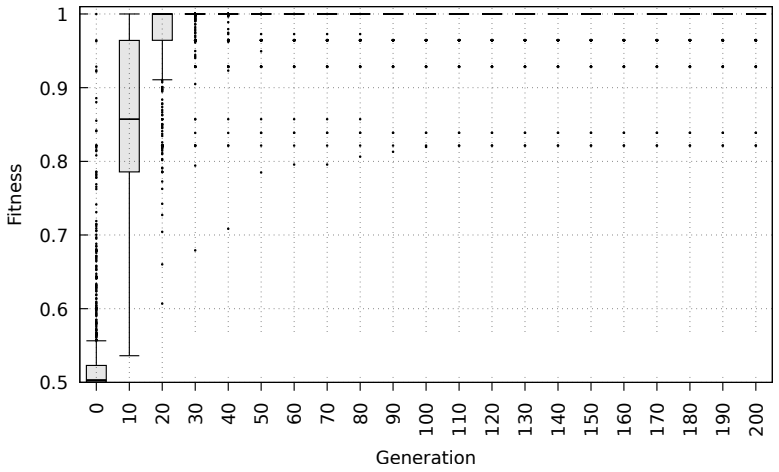


Figure 9.16: Distribution of individual fitness for 1000 DE runs for the problem of finding an RS ACCA rule that solves the DCP using only binary configurations with a density close to $\frac{1}{2}$ in the training set on the 3×3 grid.

Figure 9.16 shows the distribution of the best individual fitness in a generation for all 1000 runs for the experiment setup. We see that most of the individuals have the maximum fitness value in the 30-th generation, whereas for the previous experiments the maximum fitness was reached around the 70-th generation. This shows that the RS ACCA rules find the solution for the classical DCP much faster, or in other words the DE algorithm can reach the best fitness faster. As a result of 1000 runs, the DE found only 918 rules that solve the ACCA on the 3×3 grid in the last validation generation. When we consider also any individual that solves the classical DCP, we get 962 rules. This is visible in Figure 9.17, which presents an overview of a single DE run. An individual that solves the DCP was found in generations 30 and 40, but then the DE algorithm pointed to a different solution that has the best fitness for the training set, but not for the validation set. The LUT of the individual from generation 40 is presented in Table 9.5. An important note is that the above individual that solves the classical DCP stays in the population till the end of the experiment, but another individual is selected as a population representation. In the 50-th generation an

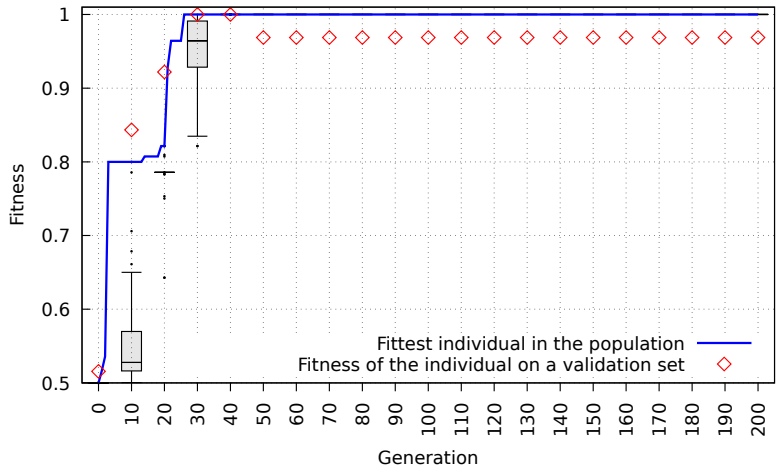


Figure 9.17: Fitness value of the best individual in a generation of an ACCA solving the DCP on the 3×3 grid where configurations with the density near $\frac{1}{2}$ and only individuals representing rotation-symmetric rules were used during evolution.

Table 9.5: A LUT of a two-dimensional rotation-symmetric ACCA rule found by the DE algorithm in 40-th generation that solves the classical DCP for all configurations on the 3×3 grid.

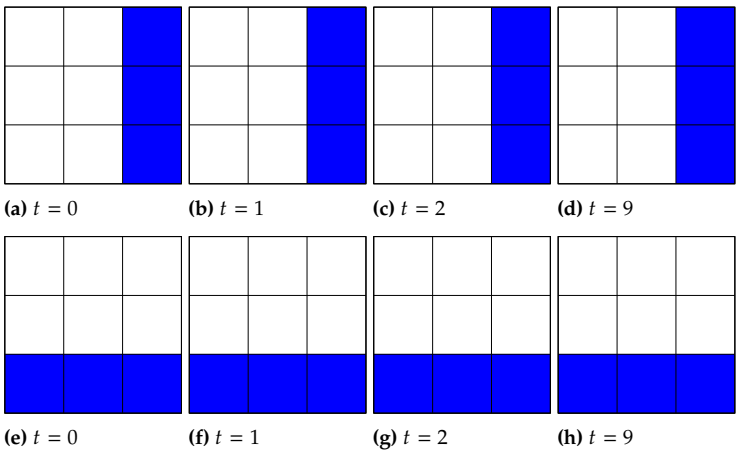
l_0	0	l_1	0	l_3	0.2616844129830059	l_4	0.1400291799791892
l_5	0.1245020844316482	l_7	0.7331455334072154	l_{10}	0.1358669631739993	l_{11}	1
l_{14}	0.8657943685452576	l_{15}	0.9741901874531294	l_{25}	1	l_{31}	1

individual presented in Table 9.6 was used for the validation process. Figure 9.18 shows a configuration evaluation that does not reach the

Table 9.6: A LUT of a two-dimensional rotation-symmetric ACCA rule found by the DE algorithm in the 40-th generation that does not solve the classical DCP for all configurations on the 3×3 grid.

l_0	0	l_1	0	l_3	0.2911521878439506	l_4	0.1740303429498948
l_5	0.0683415401066035	l_7	0.8057649329925333	l_{10}	0.0682477609104754	l_{11}	0.7092558433768025
l_{14}	1	l_{15}	0.9506689020997101	l_{25}	0.7092558433768025	l_{31}	1

required homogeneous state by using an RS ACCA rule represented by the LUT presented in Table 9.6. Those configurations represent the



(a) $t = 0$ **(b)** $t = 1$ **(c)** $t = 2$ **(d)** $t = 9$

(e) $t = 0$ **(f)** $t = 1$ **(g)** $t = 2$ **(h)** $t = 9$

Figure 9.18: Two configurations and sample iterations of an RS ACCA rule found by the DE algorithm for the 3×3 grid that does not evolve to the correct homogeneous configuration.

same, but rotated configuration. This shows yet another idea on how to reduce the configuration set just by excluding the rotation-symmetric configurations. Let us recall the $|\mathcal{I}_C| = 64$ represents all unique binary configurations and $|\mathcal{I}_T| = 28$. When we exclude rotation-symmetric configurations, our sets are reduced to 46 and 22, respectively.

Figure 9.19 shows the distribution of the LUT coefficients of RS ACCA

rules that solve the classical DCP for all binary configurations on the 3×3 grid. Like in Figure 9.9, we see the length of the interval for each

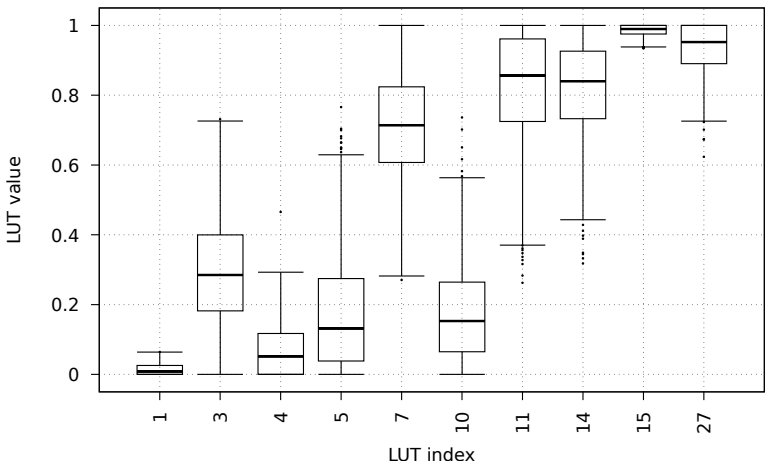


Figure 9.19: Distribution of the LUT coefficients of 962 RS ACCA rules, found in the DE experiments, that solve the classical DCP on the 3×3 grid.

coefficient is reduced.

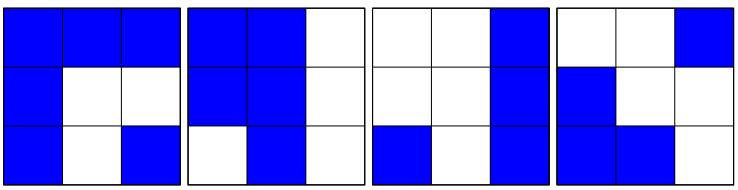
9.5.4 Observations

The above experiments presented a few challenges the DE has encountered. In a few examples, the algorithm got stuck in a local optimum, where no better individual could be found for the training set $\mathcal{I}_T$. Finding individuals by evaluating the fitness function only on a subset of initial configurations shows also how difficult it is to ensure that the individual produces also the expected evolution for other configurations.

Now, we will change the experiments a bit. We will limit the training set to RS ACCAs and configurations with the density close to $\frac{1}{2}$. The validation will only be made when a perfect individual is found in the training set. During validation, we will use all individuals that have the same, best fitness value. When an individual solves the classical DCP on a validation set, then we say the DE found the solution, thus a termination condition is met. We also modify the fitness function defined in Eq. (9.6) and introduce a threshold ϵ

$$\theta_\epsilon(\boldsymbol{x}) = \begin{cases} 1 & \text{, if } \theta_\iota(\boldsymbol{x}) \geq \epsilon \text{ ,} \\ \theta_\iota(\boldsymbol{x}) & \text{, otherwise .} \end{cases} \tag{9.8}$$

This modification returns 1 when the fitness value is greater or equal than the threshold ϵ. For our further experiments, we will use $\epsilon = 0.95$. Figure 9.20 shows a few configuration examples where an ACCA rule



(a) $t_H = 47, t_\epsilon = 4$ **(b)** $t_H = 51, t_\epsilon = 8$ **(c)** $t_H = 343, t_\epsilon = 8$ **(d)** $t_H = 341, t_\epsilon = 6$

Figure 9.20: Example of configurations whose time t_ϵ to reach the threshold value ϵ is much better (shorter) compared to the time t_H where the evolution reaches a homogeneous configuration of 0s or 1s.

from Table 9.3 was used to solve the classical DCP problem. We measured the time t_H that is required to reach a homogeneous configuration of



only 0s or 1s and the time t_e that is required to reach the threshold. We observe the imperfection of computer simulations, where the ACCA rule required only 30 iterations to reach fitness equal to 1, even if each configuration was not a fully homogeneous configuration of only 1s or 0s. The motivation for the threshold is also that the time required to classify the configuration for a threshold is shorter, thus the computational time is shorter.

In DE, there is no best parameter setup common for all optimization problems. A small population size in DE may have a negative impact resulting that the DE may converge too fast to local minima. On the other hand, when the population is too large, the time required to find the answer will be too long and will not improve the overall results. Table 9.7 presents results of experiments where a different population size $\mathcal{P}$ and termination condition for configuration iteration T were used. We see that increasing the population size and also decreasing the number of configuration evaluations have an influence on the number of experiments required to find the RS ACCAs that solve the classical DCP. The optimal setup for the experiments is $T = 10$ and $|\mathcal{P}| = 40$.

T \ $\lvert\mathcal{P}\rvert$	10	15	20	25	30	35	40	45	50
10	516	195	77	14	10	4	1	1	0
15	512	174	73	22	5	1	2	1	0
20	534	222	94	36	10	6	0	0	0
25	595	250	98	47	17	7	2	1	1
30	612	273	133	51	21	5	3	3	0
35	598	295	105	66	23	9	5	1	0
40	606	313	145	48	23	14	5	1	2
45	634	319	133	52	24	12	1	4	1
50	628	299	151	73	34	12	8	3	1

Table 9.7: Relation between population size $\mathcal{P}$ and termination condition for a number of configuration iterations T and the number of failed experiments in 1000 runs of experiments for problem of finding RS ACCA rules that solve the classical DCP on the 3×3 grid.

9.6 Experiments for a 5×3 grid

In this section all experiments will be run on a 5×3 grid where $|\mathcal{I}_C| = 2192$ and $|\mathcal{I}_T| = 858$. As we see, increasing the grid size in one direction causes the number of configurations to grow extremely.

9.6.1 Rotation-symmetric rules

We start the experiments by finding RS ACCA rules that should solve the classical DCP for the 5×3 grid. Unfortunately, we cannot cut down the set of initial configurations to rotation-symmetric ones only. We run 1000 experiments using the fitness function defined in Eq. (9.8) and use a set of parameters T and $|\mathcal{P}|$. The termination condition for the number of generations is kept at $G = 200$.

The results are presented in Table 9.8. As in the experiments for the 3×3 grid, we see that the optimal parameter setup is for $|\mathcal{P}| = 40$. Figure 9.21

Table 9.8: Relation between population size $\mathcal{P}$ and a termination condition for a number of configuration iterations T and the number of failed experiments in 1000 runs of experiments for the problem of finding RS ACCA rules that solve the classical DCP on the 5×3 grid.

T \ $\lvert\mathcal{P}\rvert$	30	35	40	45	50
20	14	8	2	1	0
25	24	5	6	0	0
30	23	11	2	1	0
35	26	9	3	1	1
40	33	10	2	2	0
45	39	9	4	3	1
50	50	13	7	2	1
55	50	16	6	0	0
60	42	17	5	1	2

shows the distribution of generations required to find the solution in relation to the population size with a termination condition $T = 20$. We

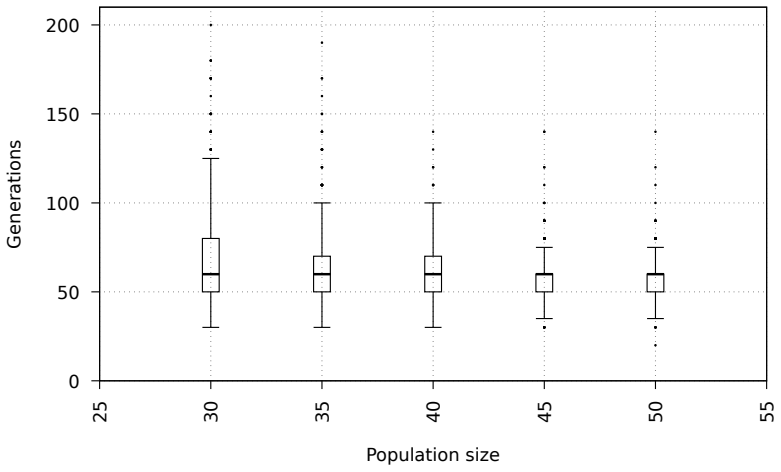


Figure 9.21: Distribution of generations in relation to the population size for the DE algorithm and a problem of finding an ACCA that solves a classical DCP for the 5×3 grid with termination condition $T = 20$.

see that the number of generations required to find a solution is lowering with the population size.

For parameters $|\mathcal{P}| = 40$ and $T = 20$, the DE algorithm found 998 RS ACCA rules that solve the classical DCP. Figure 9.22 shows the LUT coefficient distribution of all the RS ACCA rules. We see a visible

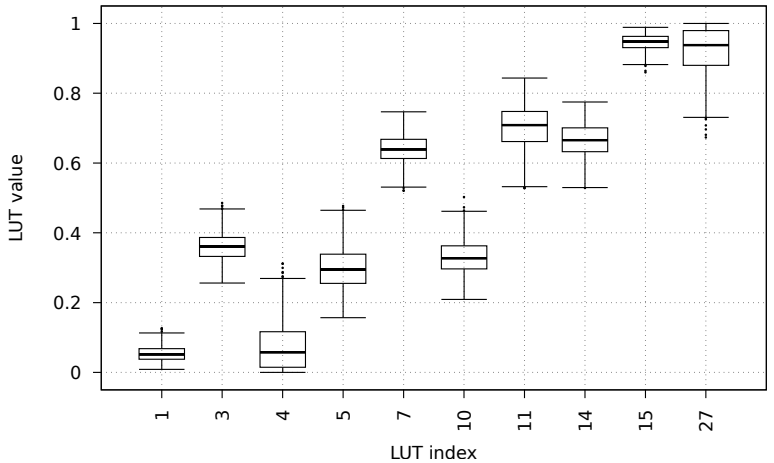


Figure 9.22: LUT coefficient distribution of the 996 best RS ACCAs found in DE experiments for the problem of finding an ACCA that solves a classical DCP for the 5×3 grid.

difference when we compare the results with the one for a 3×3 presented

in Figure 9.19, *i.e.*, the values are spread across larger ranges for the 3×3 grid.

During the 1000 DE runs, only two of them have not found any RS ACCA rules that solve the classical DCP. We present one LUT example in Table 9.9. An overview of the experiment is presented in Figure 9.23

Table 9.9: The best LUT of a two-dimensional rotation-symmetric ACCA rule found by DE algorithm that does not solve the classical DCP for all configurations on the 5×3 grid.

l_0	0	l_1	0.0358827459339008	l_3	0.3372881216812709	l_4	0
l_5	0.2482362402479622	l_7	0.5895856817806199	l_{10}	0.3117151407026815	l_{11}	0.8500075855410448
l_{14}	0.5682230879533912	l_{15}	1	l_{25}	0.8500075855410448	l_{31}	1

and shows that the fitness cannot reach a perfect solution. Figure 9.24

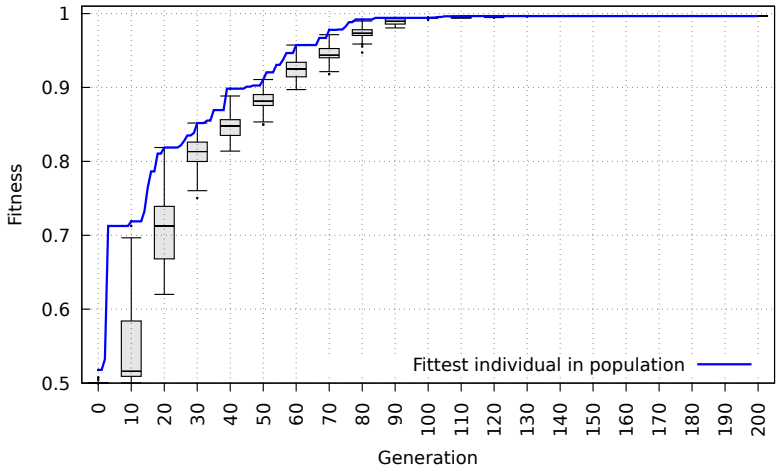


Figure 9.23: Fitness value of the best individual in a given generation of an ACCA that does not fully solve the classical DCP on the 5×3 grid.

shows all configurations that were not classified under that rule. There

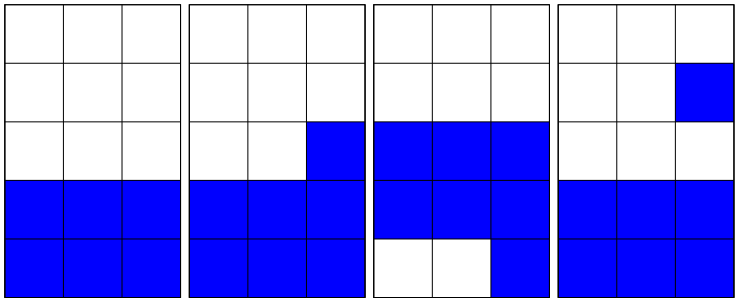


Figure 9.24: Configurations that were not classified by the fittest RS ACCA rule found by the DE algorithm for the classical DCP on the 5×3 grid.

are three configurations that were used in the training set and one that was present only in the validation set (Figure 9.24a).

9.6.2 Basin of 3×3 rules

In this section, we focus on an experiment where the initial population is a set of randomly chosen RS ACCA rules that solve the classical DCP on the 3×3 grid. As a result, for the parameter setup $|\mathcal{P}| = 40$ and $T = 10$, we build the population from 999 randomly chosen RS ACCA rules. We use $|\mathcal{P}| = \{30, 35, 40, 45, 50\}$ and $T = 20$ and run 1000 experiments, *i.e.*, we use the same parameter setup as for the rotation-symmetric case

presented before. We present the distribution of generations required to find a perfect solution in Figure 9.25. All experiments found an RS

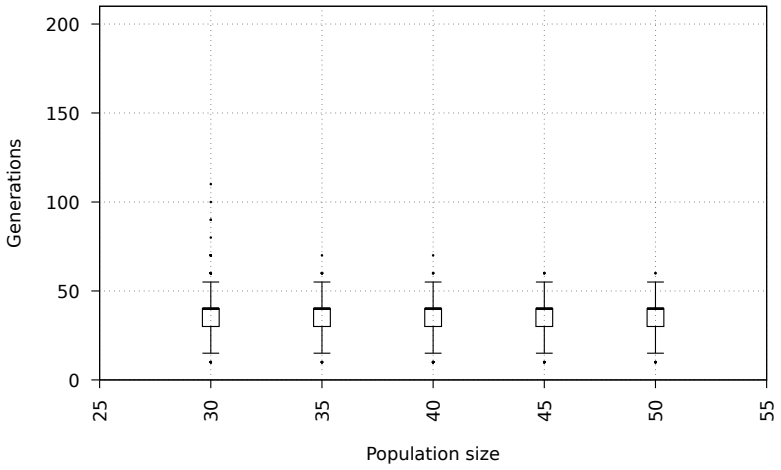


Figure 9.25: Distribution of DE generations in relation to the population size for the problem of finding an ACCA rule that solves the classical DCP for the 5×3 grid with termination condition $T = 20$ where the initial population consists of randomly chosen RS ACCA that solve the classical DCP for the 3×3 grid.

ACCA rule that solves the classical DCP for the 5×3 grid. In all DE runs, we found no solution in the initial population. This means that RS ACCA rules found for 3×3 do not solve the classical DCP for the 5×3 grid. The number of generations required to find a solution, compared to the experiment results presented in Figure 9.21, is lower.

We run the experiments one more time, but with an increased number of experiments for each pair $\mathcal{P}, T$ to 5000. Also, we used all valid RS ACCA rules that were found the 3×3 grid as a set of rules that can be used for the initial population, *i.e.*, 71723 RS ACCA rules were used it total. As a result, only one experiment did not find any solution and a few experiments found a solution in the initial population. The experiment found 24990 unique RS ACCA rules that solve the problem on the 5×3 grid.

9.6.3 Observations on smaller grids

Here we just check if the RS ACCA rules that solve the classical DCP for the 5×3 grid, solve also the problem for the 3×3 grid. For each rule, we run the same validation process as in the DE on all unique 3×3 configurations.

First, we start with the RS rules where the individuals in the initial populations were randomly selected. From the set of all RS rules, 268 were not solving the DCP for all 3×3 configurations. An in-depth

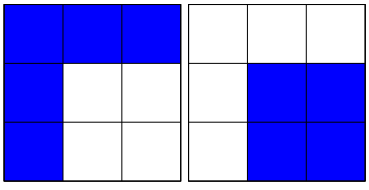


Figure 9.26: Examples of 3×3 configurations that were not classified by a RS ACCA rule that solves all configurations on the 5×3 grid.

analysis shows that only two configuration representatives presented in Figure 9.26 were not classified and only a single rule did not classify one of those two configurations.

We run the same validation for RS ACCA rules that were found in experiments where the initial population was a basin of randomly selected RS ACCA rules that solve the problem for 3×3 configurations. From a set of 24990 RS rules, only three of them do not solve the problem for the 3×3 grid and each rule cannot solve one configuration presented in Figure 9.26.

9.7 Experiments for a 5×5 grid

In this section, we use a 5×5 grid for the experiments. In this case, $|\mathcal{I}_C| = 1\,342\,210$ and $|\mathcal{I}_T| = 416\,024$. When we exclude the rotation-symmetric configurations, then our sets are reduced to 1 245 856 and 395 488 configurations, respectively.

We start immediately with an experiment where the population is selected from randomly chosen RS ACCA rules that solve the classical DCP on the 5×3 grid and 3×3 grid, *i.e.*, the initial populations are built from 24987 randomly selected rules. We run 1000 experiments for variables, where $|\mathcal{P}| = 40$ and $T \in \{20, 25, 30, 35, 40\}$ and where $G = 200$.

From the above pair we obtain results for $T \in \{30, 35, 40\}$ only, where respectively 863, 997 and 1000 experiments found ACCA rules that solve the problem for the 5×5 grid. The distribution of generations required to find the solution for the 5×5 grid compared with the number of evaluations of a single configuration is presented in Figure 9.27. We

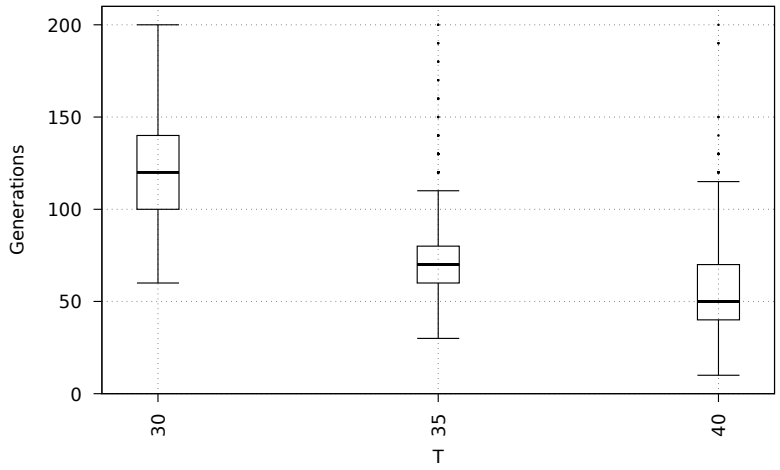


Figure 9.27: Distribution of generations for population size $|\mathcal{P}| = 40$ and termination condition $T \in \{30, 35\}$ for the DE algorithm and the problem of finding an ACCA that solves the classical DCP for the 5×5.

see that more evaluations of the configuration are required to find the answer compared with the experiments for smaller grids. For $T = 30$, some experiments found the result at the termination generation, *i.e.*, $G = 200$. An optimal setup for this grid is $T = 35$. An example of an ACCA rule that solves the DCP for all 5×5 configurations is presented in Table 9.10. All rules that solve the DCP for this grid, solve it also for

Table 9.10: The LUT of a two-dimensional rotation-symmetric ACCA rule found by DE algorithm that solves the classical DCP for all configurations on the 5×3 grid.

l_0	0	l_1	0.060328328958084	l_3	0.254681206629985	l_4	0.335276658517301
l_5	0.517163464883367	l_7	0.739827337566127	l_{10}	0.282688864840078	l_{11}	0.481679780408328
l_{14}	0.720433734455999	l_{15}	0.940914435129367	l_{25}	0.687190151552746	l_{31}	1

smaller grids. The LUT coefficient distribution of all RS ACCAs found by the DE experiments is presented in Figure 9.28

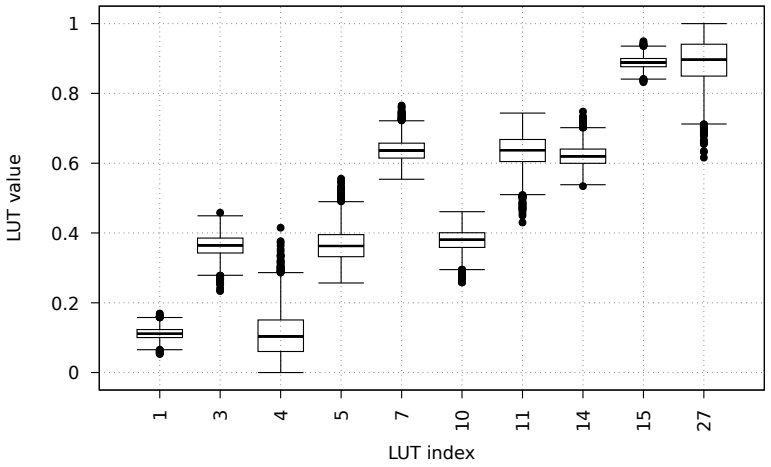


Figure 9.28: LUT coefficient distribution of the 2860 best RS ACCAs found in DE experiments for the problem of finding an ACCA that solves the classical DCP for the 5×5 grid.

9.8 Conclusions

From the experiments presented in this chapter, we see that finding ACCA rules that solve the two-dimensional DCP is a challenging, but possible task. We have used our experience gained in the one-dimensional case and, with the help of DE, we have found ACCA rules that solve the classical DCP for a given grid size. The challenge in DE is not only the problem to be solved, but also the optimization of DE parameters, such as the population size and the number of iterations that should be used in the configuration evaluation.

Few things were optimized. Focusing on rotation-symmetric rules only, as for the density-converting experiments in Chapter 8, shows that it is worth using that particular ACCA subset. This change reduced the number of individual coefficients ι_i from 30 to 10. In all our experiments, we saw that the optimal population size was $|\mathcal{P}| = 40$, but this is not a verified observation that will also work for other grid sizes. When looking for ACCA rules that would solve the problem for a particular $n \times n$ grid, in the initial population we used ACCAs that solved the problem for smaller grids. That significantly reduced the number of generations required to find the best individuals. As a result, the individuals that were solving the problem on a larger grid were also solving the problem on smaller grids - this of course is not an analytically verified fact.

At this moment, the biggest challenge for us is the verification process, where an ACCA rule is verified against any given $n \times n$ configuration. We saw that the number of configurations that should be verified grew enormously with the grid size. Focusing only on shift-invariant configurations as mentioned in Section 2.3 would just slightly reduce the required computations. For the next grid 7×7, we got around $11.5 \cdot 10^{12}$ unique configurations to verify. Reducing the set to configurations where $\rho(\boldsymbol{x}) \approx \frac{1}{2}$, reduced the configuration set to only around $2.58 \cdot 10^{12}$. As we saw on the 3×3 grid, learning only on the set of configurations with the density near to $\frac{1}{2}$ resulted in a situation where a particular configuration beyond the set could not be classified. This part can be improved by

verifying configurations and classifying them into different groups of complexity. Also in our experiments, we were not focusing on the number of iterations required to classify a configuration, as that was not our optimization problem. In Section 9.5.4, we changed the fitness function, as waiting for a perfect homogeneous configuration with all 1s or 0s is resource intensive. Further experiments and analytical research in this area, similarly as in Section 7.3, may help to find terminating conditions for T.

The above observations may suggest a few hypotheses that should be checked in detail and verified:

- There is a two-dimensional ACCA rule that solves the classical DCP for two-dimensional $n \times n$ configurations (and for smaller grids), where n is odd.
- There is a certain subset of configurations that should always be included in the training set.
- There is a terminating condition for examining configuration classification.
- There is a set of optimal parameters for the DE and the DCP.

Conclusion 10

10.1 Summary 131
10.2 Future work 132

This work concerned the Density Classification Problem, one of the best known decision problems in the field of CAs. We focused on a special type of Continuous CAs, Affine Continuous CAs, in the context of solving this problem. With the help of computer simulations, it was possible to discover and analytically prove a few interesting observations for ACCAs. In this chapter, we summarize the work and also present topics that exceed it but are worth investigating.

10.1 Summary

This work presented the results of the investigation of ACCAs, the simplest generalization of ECAs to CCAs, with a focus on ACCAs in the context of the DCP.

Chapter 4 presented a negative outcome stating that there is no ACCA that solves the DCP in the classical sense. As that proof could end the research on that topic, in Chapter 5 a complete description of the dynamics of legal outer-totalistic ACCAs was given. From that point it has become clear that ACCAs, being the simplest generalization of ECAs, constitute a set of CAs with very interesting properties, which are not observable in the world of ECAs. This discovery provided motivation for further investigation in the mentioned topic. The results presented in Chapter 5 also show how difficult the computations related to ACCAs can be.

By relaxing the classical DCP definition by a small modification of the output specification, in Chapter 6, we presented the first results for one-dimensional ACCAs. The results with analytical proofs state that almost each element of the class of density-conserving ACCAs is able to solve the DCP in the relaxed sense. An extension of these results for two-dimensional ACCAs was presented in Chapter 8 where on the one hand, it was shown that those simulations and proofs are much more challenging and, on the other hand, that it is sufficient to examine just a subspace of rotation-symmetric rules to get interesting results.

As there is no ACCA that solves the DCP in the classical sense for all grids, it makes sense to check the classical DCP for fixed-length configurations. Indeed, in contrast to ECAs, ACCAs can solve this version of DCP. For the one-dimensional case, Chapter 7 presented the simulations with proofs of the obtained results. The research for two-dimensional ACCAs, presented in Chapter 9, is much more difficult. However, using differential evolution algorithms and the knowledge gained from previous chapters, it was possible to explore a wide range of ACCAs that solve the DCP for fixed-length two-dimensional grids.

Even after exhaustive simulations and analytical proofs, there are still interesting questions for further work.

10.2 Future work

The research presented in this work can be extended in many directions. Below, just some of them and open questions worth investigation are outlined.

When considering the one-dimensional case, only the neighborhood of radius 1 was used. As a continuation of the research, ACCAs with a larger radius can be studied. In particular all definitions presented in this work can be easily reformulated for larger radii. It is also possible to go beyond the binary set of states. Unfortunately, this change requires quite a lot of interference in the basic definitions, *i.e.*, the definition of ACCA. However, such considerations have already appeared in the literature as q-VFCA: q-state Vector-valued Fuzzy Cellular Automata [88, 89].

In the two-dimensional case only the classical version of the von Neumann neighborhood was used. However, one can consider the von Neumann neighborhood with larger radii, even the Moore neighborhood or other ones [90]. We should remember that the choice of the von Neumann neighborhood not only affected the reduction of computational time but also simplified analytical proofs, *i.e.*, we see that in the formulation of LUT tables. On the other hand, changing the neighborhood, for example, to the Moore neighborhood will impede the analytical proofs, but potentially may give also interesting results. At least simulations could be run for that type of neighborhood.

Considering the two-dimensional case it is worth to recall that there are other than a rectangular tessellation of the plane. One can use the hexagonal [32, 34, 91] or the triangular [91, 92] grid. In addition, irregular divisions of the plane can be considered, as for example in Voronoi spatial models [93, 94].

The exploration and analysis of the classical DCP for two-dimensional ACCAs is still an open question, as analytical proofs are required.

So far, no studies have been carried out for three-dimensional ACCAs, even with the von Neumann neighborhood. The results for such ACCAs should be interesting, especially since it is proven that regardless of the dimension d, binary number-conserving CAs with the von Neumann neighborhood are trivial. They are intrinsically one-dimensional [95], while the generalization of these CAs, *i.e.*, ACCAs do not reduce to one-dimensional (as it can be seen in the two-dimensional case).

There are also many "technical" aspects that can be worked on in the next stage of research. For example, we saw that extending the configuration size by only a single cell doubled the number of configurations that should be evaluated in simulations. As a consequence, it is very important to reduce the set of configurations, by using for example, only those with the density close to 0.5 or by using rotation-invariant configurations only. However, those sets are still huge for a small 7×7 grid. The important question is how to reduce the time spent on simulations, *i.e.*, on what configurations we should focus.

In the simulations we used double precision floating point numbers, which are not perfect for multiplications and can give wrong results for larger grids. In the future investigation one can also try to limit to ACCAs where only rational numbers can be used in the LUT tables.

Supporting the research with any kind of EC algorithms or machine learning definitely gives interesting results and improves the research [96–98]. As mentioned in Chapter 9, we can try to find an optimal set of parameters for the DE for a given size of the configuration.

Finally, further considerations may also concern the use of ACCAs to solve other decision problems, as for example, the well-known parity problem [99, 100].

Bibliography

[1] B. Wolnik, M. Dembowski, W. Bołt, J. M. Baetens, and B. De Baets. *The Density Classification Problem in the Context of Continuous Cellular Automata*. In: *Cellular Automata: 12th International Conference on Cellular Automata for Research and Industry, ACRI 2016, Fez, Morocco, September 5-8, 2016. Proceedings*. Ed. by S. El Yacoubi, J. Wąs, and S. Bandini. Cham: Springer International Publishing, 2016, pp. 79–87. DOI: 10.1007/978-3-319-44365-2_8 (cited on page iii).

[2] B. Wolnik, M. Dembowski, W. Bołt, J. M. Baetens, and B. De Baets. *Density-conserving affine continuous cellular automata solving the relaxed density classification problem*. In: *Journal of Physics A: Mathematical and Theoretical* 50.34 (2017), p. 345103. DOI: 10.1088/1751-8121/aa7d86 (cited on pages iii, 88).

[3] M. Dembowski, B. Wolnik, W. T. Bołt, J. Baetens, and B. De Baets. *Affine continuous cellular automata solving the fixed-length density classification problem*. In: *Natural Computing* 17.3 (2018), pp. 467–477. DOI: 10.1007/s11047-017-9631-4 (cited on page iii).

[4] M. Dembowski, B. Wolnik, W. T. Bołt, J. Baetens, and B. De Baets. *Two-dimensional affine continuous cellular automata solving the relaxed density classification problem*. In: *Journal of Cellular Automata* 14.3-4 (2019), pp. 191–212 (cited on page iii).

[5] B. Wolnik, M. Dembowski, A. Augustynowicz, and B. De Baets. *A complete description of the dynamics of legal outer-totalistic affine continuous cellular automata*. In: *Nonlinear Dynamics* 110.1 (2022), pp. 589–610. DOI: 10.1007/s11071-022-07642-w (cited on page iii).

[6] M. Gardner. *Mathematical Games - The fantastic combinations of John Conway's new solitaire game "life"*. In: *Scientific American* 223 (1970), pp. 120–123. DOI: 10.1038/scientificamerican1070-120 (cited on pages 2, 41).

[7] J. von Neumann. *The general and logical theory of automata.* In: Cerebral Bechanisms in Behavior; the Hixon Symposium. Oxford, England: Wiley, 1951, pp. 1–41 (cited on page 2).

[8] J. V. Neumann and A. W. Burks. *Theory of Self-Reproducing Automata*. USA: University of Illinois Press, 1966 (cited on page 2).

[9] K. Zuse. *Rechnender Raum (Calculating Space)*. In: *Schriften Zur Dataverarbeitung* 1 (1969) (cited on page 2).

[10] J. M. Greenberg and S. P. Hastings. *Spatial Patterns for Discrete Models of Diffusion in Excitable Media*. In: *SIAM Journal on Applied Mathematics* 34.3 (1978), pp. 515–523. DOI: 10.1137/0134040 (cited on page 3).

[11] M. Cook. *Universality in Elementary Cellular Automata*. In: *Complex Systems* 15 (Jan. 2004) (cited on pages 3, 16).

[12] *A New Kind of Science*. Champaign, Ilinois, USA: Wolfram Media Inc., 2002 (cited on pages 3, 15).

[13] L. Gray. *A mathematician looks at S. Wolfram's new kind of science*. In: *Notices of the American Mathematical Society* 50 (2003) (cited on page 3).

[14] G. t Hooft. *The Cellular Automaton Interpretation of Quantum Mechanics. A View on the Quantum Nature of our Universe, Compulsory or Impossible?* In: (May 2014) (cited on page 3).

[15] *Published Papers Per Year on Cellular Automata*. 2015. DOI: 10.24097/wolfram.44993.data (cited on page 3).

[16] S. Wolfram. *Random sequence generation by cellular automata*. In: *Advances in Applied Mathematics* 7.2 (986), pp. 123–169. DOI: 10.1016/0196-8858(86)90028-X (cited on page 4).

[17] S. Wolfram. *Cryptography with Cellular Automata*. In: *Advances in Cryptology — CRYPTO '85 Proceedings*. Ed. by H. C. Williams. Berlin, Heidelberg: Springer Berlin Heidelberg, 1986, pp. 429–432 (cited on page 4).

[18] M. Tomassini and M. Perrenoud. *Cryptography with cellular automata*. In: *Applied Soft Computing* 1.2 (2001), pp. 151–160. DOI: 10.1016/S1568-4946(01)00015-1 (cited on page 4).

[19] S. Nandi, B. K. Kar, and P. Pal Chaudhuri. *Theory and applications of cellular automata in cryptography*. In: *IEEE Transactions on Computers* 43.12 (1994), pp. 1346–1357 (cited on page 4).

[20] Z. Liu, Y. Wang, Y. Zhao, and L. Y. Zhang. *A stream cipher algorithm based on 2D coupled map lattice and partitioned cellular automata*. In: *Nonlinear Dynamics* 101.2 (July 2020), pp. 1383–1396. DOI: 10.1007/s11071-020-05804-2 (cited on page 4).

[21] X. Lv, X. Liao, and B. Yang. *A novel pseudo-random number generator from coupled map lattice with time-varying delay*. In: *Nonlinear Dynamics* 94.1 (Oct. 2018), pp. 325–341. DOI: 10.1007/s11071-018-4361-4 (cited on page 4).

[22] B. Płaczek. *A Traffic Model Based on Fuzzy Cellular Automata*. In: *Journal of Cellular Automata* 8 (Nov. 2013) (cited on page 4).

[23] S. Maerivoet and B. De Moor. *Cellular automata models of road traffic*. In: *Physics Reports* 419.1 (Nov. 2005), pp. 1–64. DOI: 10.1016/j.physrep.2005.08.005 (cited on page 4).

[24] K. Nagel and M. Schreckenberg. *A cellular automaton model for freeway traffic*. In: *Journal de Physique I* 2 (Dec. 1992), p. 2221 (cited on page 4).

[25] W. Yuan and K. H. Tan. *A model for simulation of crowd behaviour in the evacuation from a smoke-filled compartment*. In: *Physica A: Statistical Mechanics and its Applications* 390.23 (2011), pp. 4210–4218. DOI: 10.1016/j.physa.2011.07.044 (cited on page 4).

[26] Y. Xu and H.-J. Huang. *Simulation of exit choosing in pedestrian evacuation with consideration of the direction visual field*. In: *Physica A: Statistical Mechanics and its Applications* 391.4 (2012), pp. 991–1000. DOI: 10.1016/j.physa.2011.10.022 (cited on page 4).

[27] J. Wąs and R. Lubaś. *Adapting Social Distances Model for Mass Evacuation Simulation*. In: *Journal of Cellular Automata* (Nov. 2013) (cited on page 4).

[28] R. Lovreglio, E. Ronchi, and D. Nilsson. *Calibrating floor field cellular automaton models for pedestrian dynamics by using likelihood function optimization*. In: *Physica A: Statistical Mechanics and its Applications* 438 (2015), pp. 308–320. DOI: 10.1016/j.physa.2015.06.040 (cited on page 4).

[29] K. Zhu, Y. Yang, and Q. Shi. *Study on evacuation of pedestrians from a room with multi-obstacles considering the effect of aisles*. In: *Simulation Modelling Practice and Theory* 69 (2016), pp. 31–42. DOI: 10.1016/j.simpat.2016.09.002 (cited on page 4).

[30] R. Lubaś, J. Porzycki, J. Wąs, and M. Mycek. *Validation and Verification of CA-Based Pedestrian Dynamics Models*. In: *Journal of Cellular Automata* 11 (July 2016), pp. 285–298 (cited on page 4).

[31] Z. Fu, Q. Jia, J. Chen, J. Ma, K. Han, and L. Luo. *A fine discrete field cellular automaton for pedestrian dynamics integrating pedestrian heterogeneity, anisotropy, and time-dependent characteristics*. In: *Transportation Research Part C: Emerging Technologies* 91 (2018), pp. 37–61. DOI: 10.1016/j.trc.2018.03.022 (cited on page 4).

[32] A. T. Nugraha, B. J. Waterson, S. P. Blainey, and F. J. Nash. *On the consistency of urban cellular automata models based on hexagonal and square cells*. In: *Environment and Planning B: Urban Analytics and City Science* 48.4 (2021), pp. 845–860. DOI: 10.1177/2399808319898501 (cited on pages 4, 132).

[33] A. Varas, M. Cornejo, D. Mainemer, B. Toledo, J. Rogan, V. Muñoz, and J. Valdivia. *Cellular automaton model for evacuation process with obstacles*. In: *Physica A: Statistical Mechanics and its Applications* 382.2 (2007), pp. 631–642. DOI: 10.1016/j.physa.2007.04.006 (cited on pages 4, 5).

[34] S. Wolfram. *Cellular automaton fluids 1: Basic theory*. In: *Journal of Statistical Physics* 45.3 (Nov. 1986), pp. 471–526. DOI: 10.1007/BF01021083 (cited on pages 4, 5, 132).

[35] G. Ortigoza, F. Brauer, and I. Neri. *Modelling and simulating Chikungunya spread with an unstructured triangular cellular automata*. In: *Infectious Disease Modelling* 5 (2020), pp. 197–220. DOI: 10.1016/j.idm.2019.12.005 (cited on pages 4, 5).

[36] D. Makowiec, W. Miklaszewski, J. Wdowczyk, and A. T. Lawniczak. *From cellular automata model of vagal control of the human right atrium to heart beats patterns*. In: *Physica D: Nonlinear Phenomena* 415 (2021), p. 132771. DOI: 10.1016/j.physd.2020.132771 (cited on pages 4, 5).

[37] T. Komatsuzaki, Y. Iwata, and S. Morishita. *Modelling of Incident Sound Wave Propagation around Sound Barriers Using Cellular Automata*. In: *Cellular Automata*. Ed. by G. C. Sirakoulis and S. Bandini. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 385–394 (cited on page 4).

[38] V. V. Nishawala, M. Ostoja-Starzewski, M. J. Leamy, and P. N. Demmie. *Simulation of elastic wave propagation using cellular automata and peridynamics, and comparison with experiments*. In: *Wave Motion* 60 (2016), pp. 73–83. DOI: 10.1016/j.wavemoti.2015.08.005 (cited on page 4).

[39] L. H. Encinas, S. H. White, A. M. del Rey, and G. R. Sánchez. *Modelling forest fire spread using hexagonal cellular automata*. In: *Applied Mathematical Modelling* 31.6 (2007), pp. 1213–1227. DOI: 10.1016/j.apm.2006.04.001 (cited on page 4).

[40] J. G. Freire and C. C. DaCamara. *Using cellular automata to simulate wildfire propagation and to assist in fire management*. In: *Natural Hazards and Earth System Sciences* 19.1 (2019), pp. 169–179. DOI: 10.5194/nhess-19-169-2019 (cited on page 4).

[41] J. Czerniak, H. Zarzycki, L. Apiecionek, W. Palczewski, and P. Kardasz. *A Cellular Automata-Based Simulation Tool for Real Fire Accident Prevention*. In: *Mathematical Problems in Engineering* 2018 (Feb. 2018), pp. 1–12. DOI: 10.1155/2018/3058241 (cited on page 4).

[42] S. H. White, A. M. del Rey, and G. R. Sánchez. *Modeling epidemics using cellular automata*. In: *Applied Mathematics and Computation* 186.1 (2007), pp. 193–202. DOI: 10.1016/j.amc.2006.06.126 (cited on page 4).

[43] D. Makowiec, J. Wdowczyk, and Z. R. Struzik. *Heart Rhythm Insights Into Structural Remodeling in Atrial Tissue: Timed Automata Approach*. In: *Frontiers in Physiology* 9 (2019), p. 1859. DOI: 10.3389/fphys.2018.01859 (cited on page 4).

[44] R. S. Campos, R. M. Amorim, B. L. de Oliveira, B. M. Rocha, J. Sundnes, L. P. da Silva Barra, M. Lobosco, and R. W. dos Santos. *3D Heart Modeling with Cellular Automata, Mass-Spring System and CUDA*. In: *Parallel Computing Technologies*. Ed. by V. Malyshkin. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 296–309 (cited on page 4).

[45] N. Fatès, É. Thierry, M. Morvan, and N. Schabanel. *Fully asynchronous behavior of double-quiescent elementary cellular automata*. In: *Theoretical Computer Science* 362.1 (2006), pp. 1–16. DOI: 10.1016/j.tcs.2006.05.036 (cited on pages 13, 84).

[46] G. Cattaneo, P. Flocchini, G. Mauri, C. Vogliotti, and N. Santoro. *Cellular automata in fuzzy backgrounds*. In: *Physica D: Nonlinear Phenomena* 105.1 (1997), pp. 105–120. DOI: 10.1016/S0167-2789(96)00233-3 (cited on page 17).

[47] P. Flocchini, F. Geurts, A. Mingarelli, and N. Santoro. *Convergence and aperiodicity in fuzzy cellular automata: revisiting rule 90*. In: *Physica D: Nonlinear Phenomena* 142.1–2 (2000), pp. 20–28. DOI: 10.1016/S0167-2789(00)00052-X (cited on page 17).

[48] H. Betel and P. Flocchini. *On the asymptotic behavior of fuzzy cellular automata*. In: *Electronic Notes in Theoretical Computer Science* 252 (2009), pp. 23–40. DOI: 10.1016/j.entcs.2009.09.012 (cited on pages 17, 19, 23, 72).

[49] H. Betel and P. Flocchini. *On the Relationship Between Boolean and Fuzzy Cellular Automata*. In: *Electronic Notes in Theoretical Computer Science* 252 (2009), pp. 5–21. DOI: 10.1016/j.entcs.2009.09.011 (cited on pages 17, 19, 27, 31, 36).

[50] P. Flocchini and V. Cezar. *Radial View of Continuous Cellular Automata*. In: *Fundamenta Informaticae* 87.2 (2008), pp. 165–183 (cited on page 18).

[51] K. Kaneko. *Theory and Applications of Coupled Map Lattices*. Wiley, 1993 (cited on page 18).

[52] G. Cattaneo, P. Flocchini, G. Mauri, C. Vogliotti, and N. Santoro. *Cellular automata in fuzzy backgrounds*. In: *Physica D: Nonlinear Phenomena* 105.1–3 (1997), pp. 105–120 (cited on page 19).

[53] N. Boccara and H. Fukundefined. *Number-Conserving Cellular Automaton Rules*. In: *Fundamenta Informaticae* 52.1–3 (2002), pp. 1–13 (cited on page 23).

[54] H. Fukś. *Probabilistic cellular automata with conserved quantities*. In: *Nonlinearity* 17.1 (2004), p. 159 (cited on page 24).

[55] B. Wolnik, A. Dzedzej, J. M. Baetens, and B. De Baets. *Number-conserving cellular automata with a von Neumann neighborhood of range one*. In: *Journal of Physics A: Mathematical and Theoretical* 50.43 (2017), p. 435101 (cited on page 26).

[56] B. Durand, E. Formenti, and Z. Róka. *Number-conserving cellular automata I: decidability*. In: *Theoretical Computer Science* 299.1 (2003), pp. 523–535. DOI: 10.1016/S0304-3975(02)00534-0 (cited on page 26).

[57] B. Wolnik, A. Nenca, J. M. Baetens, and B. De Baets. *A split-and-perturb decomposition of number-conserving cellular automata*. In: *Physica D: Nonlinear Phenomena* 413 (2020), p. 132645. DOI: 10.1016/j.physd.2020.132645 (cited on page 28).

[58] P. Gács, G. L. Kurdyumov, and L. A. Levin. *One-dimensional uniform arrays that wash out finite islands*. In: *Problemy Peredachi Informatsii* 14.3 (1978), pp. 92–96 (cited on page 33).

[59] N. H. Packard. "Adaptation toward the edge of chaos". In: *Dynamic Patterns in Complex Systems*. Ed. by J. A. S. Kelso, A. J. Mandell, and M. F. Shlesinger. World Scientific, 1988, pp. 293–301 (cited on page 33).

[60] M. Land and R. K. Belew. *No perfect two-state cellular automata for density classification exists*. In: *Physical Review Letters* 74 (25 1995), pp. 5148–5150 (cited on pages 33, 34).

[61] M. S. Capcarrère and M. Sipper. *Necessary conditions for density classification by cellular automata*. In: *Physical Review E* 64 (3 Aug. 2001), p. 036113. DOI: 10.1103/PhysRevE.64.036113 (cited on page 33).

[62] M. Mitchell, J. P. Crutchfield, and R. Das. *Evolving cellular automata with genetic algorithms: A review of recent work*. In: *Proceedings of the First International Conference on Evolutionary Computation and Its Applications (EvCA'96)*. 1996 (cited on page 33).

[63] P. P. de Oliveira, J. C. Bortot, and G. M. Oliveira. *The best currently known class of dynamically equivalent cellular automata rules for density classification*. In: *Neurocomputing* 70.1–3 (2006). Neural NetworksSelected Papers from the 7th Brazilian Symposium on Neural Networks (SBRN '04)7th Brazilian Symposium on Neural Networks, pp. 35–43. DOI: 10.1016/j.neucom.2006.07.003 (cited on page 33).

[64] H. Fukś. *Solution of the density classification problem with two cellular automata rules*. In: *Physical Review E* 55 (3 1997), R2081–R2084 (cited on page 33).

[65] C. Stone and L. Bull. *Solving the density classification task using cellular automaton 184 with memory*. In: *Complex Systems* 18.3 (2009), p. 329 (cited on page 33).

[66] H. Fukś. *Nondeterministic density classification with diffusive probabilistic cellular automata*. In: *Physical Review E* 66 (6 Dec. 2002), p. 066106. DOI: 10.1103/PhysRevE.66.066106 (cited on page 33).

[67] M. Schüle, T. Ott, and R. Stoop. "Computing with Probabilistic Cellular Automata". In: *Artificial Neural Networks – ICANN 2009: 19th International Conference, Limassol, Cyprus, September 14-17, 2009, Proceedings, Part II*. Ed. by C. Alippi, M. Polycarpou, C. Panayiotou, and G. Ellinas. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 525–533. DOI: 10.1007/978-3-642-04277-5_53 (cited on page 33).

[68] N. Fatès. *Stochastic cellular automata solutions to the density classification problem - when randomness helps computing*. In: *Theory of Computing Systems* 53.2 (2013), pp. 223–242. DOI: 10.1007/s00224-012-9386-3 (cited on page 33).

[69] M. S. Capcarrère, M. Sipper, and M. Tomassini. *Two-state, $r = 1$ Cellular Automaton that Classifies Density*. In: *Physical Review Letters* 77 (24 1996), pp. 4969–4971 (cited on page 33).

[70] R. Briceño, P. M. de Espanés, A. Osses, and I. Rapaport. *Solving the density classification problem with a large diffusion and small amplification cellular automaton*. In: *Physica D: Nonlinear Phenomena* 261 (2013), pp. 70–80. DOI: 10.1016/j.physd.2013.07.002 (cited on page 34).

[71] M. S. Capcarrère and M. Sipper. *Necessary conditions for density classification by cellular automata*. In: *Physical Review E* 64 (3 2001), p. 036113 (cited on page 34).

[72] A. Bušić, N. Fatès, J. Mairesse, and I. Marcovici. *Density classification on infinite lattices and trees*. In: *Electronic Journal of Probability* 18 (2013), 22 pp. DOI: 10.1214/EJP.v18-2325 (cited on page 34).

[73] D. Eppstein. "Growth and decay in Life-Like Cellular Automata". In: *Game of Life Cellular Automata*. Ed. by A. Adamatzky. London: Springer, 2010, pp. 71–97 (cited on page 41).

[74] A. Berlanga, P. Isasi, A. Sanchis, and J. M. Molina. *Uniform Coevolution for Solving the Density Classification Problem in Cellular Automata*. In: *Proceedings of the 2nd Annual Conference on Genetic and Evolutionary Computation*. GECCO'00. Las Vegas, Nevada: Morgan Kaufmann Publishers Inc., 2000, pp. 383–383 (cited on page 81).

[75] A. Dennunzio, E. Formenti, and M. Weiss. *Multidimensional Cellular Automata: Closing Property, Quasi-expansivity, and (Un)Decidability Issues*. In: *Theoretical Computer Science* 516 (Jan. 2014), pp. 40–59. DOI: 10.1016/j.tcs.2013.11.005 (cited on page 97).

[76] N. Fatès, I. Marcovici, and S. Taati. *Two-Dimensional Traffic Rules and the Density Classification Problem*. In: *Cellular Automata and Discrete Complex Systems*. Ed. by M. Cook and T. Neary. Cham: Springer International Publishing, 2016, pp. 135–148 (cited on page 97).

[77] D. C. Ince. *Mechanical Intelligence (Collected Works of A. M. Turing)*. North-Holland Publishing Co., 1992 (cited on page 106).

[78] T. Back, D. B. Fogel, and Z. Michalewicz. *Handbook of Evolutionary Computation*. IOP Publishing Ltd., 1997 (cited on page 106).

[79] R. M. Friedberg. *A Learning Machine: Part I*. In: *IBM Journal of Research and Development* 2 (1958), pp. 2–13 (cited on page 106).

[80] J. H. Holland. *Outline for a Logical Theory of Adaptive Systems*. In: *Journal of the ACM* 9.3 (July 1962), pp. 297–314. DOI: 10.1145/321127.321128 (cited on page 107).

[81] J. Holland. *Adaptation in Natural and Artificial Systems: an Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence*. University of Michigan Press, 1975 (cited on page 107).

[82] L. Fogel, A. Owens, and M. Walsh. *Artificial Intelligence through Simulated Evolution*. 1966 (cited on page 107).

[83] I. Rechenberg. *Evolutionsstrategie : Optimierung technischer Systeme nach Prinzipien der biologischen Evolution*. ger. Stuttgart : Frommann-Holzboog, 1973 (cited on page 107).

[84] H.-P. Schwefel. *Numerische Optimierung von Computermodellen mittels der Evolutionsstrategie*. Vol. 26. 1977 (cited on page 107).

[85] H.-G. Beyer and H.-P. Schwefel. *Evolution strategies - A comprehensive introduction*. In: *Natural Computing* 1 (Mar. 2002), pp. 3–52. DOI: 10.1023/A:1015059928466 (cited on page 107).

[86] T. Bäck, G. Rudolph, and H.-P. Schwefel. *Evolutionary Programming and Evolution Strategies: Similarities and Differences*. In: *Proceedings of the Second Annual Conference on Evolutionary Programming* (Apr. 1997) (cited on page 107).

[87] K. Price, R. Storn, and J. Lampinen. *Differential Evolution-A Practical Approach to Global Optimization*. Vol. 141. Jan. 2005 (cited on page 109).

[88] Y. Nishida, S. Watanabe, A. Fukuda, and Y. Watanabe. *q-VFCA: q-state Vector-valued Fuzzy Cellular Automata*. In: *Journal of Cellular Automata* 15.3 (2020), pp. 207–222 (cited on page 132).

[89] Y. Nishida, S. Watanabe, A. Fukuda, and Y. Watanabe. *Traffic flow models with two kinds of vehicles in terms of the vector-valued cellular automata and their fuzzification*. In: *AIP Conference Proceedings*. Vol. 2641. Dec. 2022. DOI: 10.1063/5.0114966 (cited on page 132).

[90] D. A. Zaitsev. *A generalized neighborhood for cellular automata*. In: *Theoretical Computer Science* 666 (2017), pp. 21–35. DOI: 10.1016/j.tcs.2016.11.002 (cited on page 132).

[91] C. Bays. *Cellular Automata in Triangular, Pentagonal and Hexagonal Tessellations*. In: *Encyclopedia of Complexity and Systems Science*. Ed. by R. A. Meyers. New York, NY: Springer New York, 2009, pp. 892–900. DOI: 10.1007/978-0-387-30440-3_58 (cited on page 132).

[92] M. Zawidzki. *Application of Semitotalistic 2D Cellular Automata on a Triangulated 3D Surface*. In: *International Journal of Design & Nature and Ecodynamics* 6 (Jan. 2011), pp. 34–51. DOI: 10.2495/DNE-V6-N1-34-51 (cited on page 132).

[93] A. Flache and R. Hegselmann. *Do Irregular Grids Make a Difference? Relaxing the Spatial Regularity Assumption in Cellular Models of Social Dynamics*. In: *Journal of Artificial Societies and Social Simulation* 4.4 (2001), pp. 1–6 (cited on page 132).

[94] A. H. F. Navid and A. B. Aghababa. "Cellular Learning Automata and Its Applications". In: *Emerging Applications of Cellular Automata*. Ed. by A. Salcido. Rijeka: IntechOpen, 2013. Chap. 5. DOI: 10.5772/52953 (cited on page 132).

[95] B. Wolnik and B. De Baets. *All binary number-conserving cellular automata based on adjacent cells are intrinsically one-dimensional*. In: *Physical Review E* 100 (2 Aug. 2019). DOI: 10.1103/PhysRevE.100.022126 (cited on page 132).

[96] *Growing Isotropic Neural Cellular Automata*. Vol. ALIFE 2022: The 2022 Conference on Artificial Life. ALIFE 2023: Ghost in the Machine: Proceedings of the 2023 Artificial Life Conference. July 2022, p. 65. DOI: 10.1162/isal_a_00552 (cited on page 133).

[97] J. M. Springer and G. T. Kenyon. *It's Hard for Neural Networks to Learn the Game of Life*. In: *2021 International Joint Conference on Neural Networks (IJCNN)*. 2021, pp. 1–8. DOI: 10.1109/IJCNN52387.2021.9534060 (cited on page 133).

[98] *Convolutional Neural Networks for Cellular Automata Classification*. Vol. ALIFE 2019: The 2019 Conference on Artificial Life. ALIFE 2023: Ghost in the Machine: Proceedings of the 2023 Artificial Life Conference. July 2019, pp. 280–281. DOI: 10.1162/isal_a_00175 (cited on page 133).

[99] H. Betel, P. P. Oliveira, and P. Flocchini. *Solving the Parity Problem in One-Dimensional Cellular Automata*. In: *Natural Computing: An International Journal* 12.3 (2013), pp. 323–337. DOI: 10.1007/s11047-013-9374-9 (cited on page 133).

[100] P. P. Balbi, E. Ruivo, and F. Faria. *Synchronous solution of the parity problem on cyclic configurations, with elementary cellular automaton rule 150, over a family of directed, non-circulant, regular graphs*. In: *Information Sciences* 615 (2022), pp. 578–603. DOI: 10.1016/j.ins.2022.10.045 (cited on page 133).

Notation

Notation used in this work.

$\boldsymbol{\iota}$	Individual, p. 105	
$\boldsymbol{x}$	Configuration, p. 8	
$\mathbb{F}$	Global rule, p. 11	
$\mathbb{f}$	Local rule, p. 11	
$\mathbb{f}^C$	Conjugation of rule $\mathbb{f}$. 14	
$\mathbb{f}^I$	Inversion of rule $\mathbb{f}$. 14	
$\mathbb{f}_d$	Diffusion rule, p. 17	
$\mathbb{N}$	Set of natural numbers	$0, 1, 2, \ldots$
$\mathbb{R}$	Set of real numbers	
$\mathbb{Z}$	Set of integer numbers	$\ldots, -2, -1, 0, 1, 2, \ldots$
$\mathcal{C}$	Cellular space (the grid), p. 7	
$\mathcal{I}_C$	Set of all binary configurations, p. 9	
$\mathcal{I}_T$	Training set of configurations, p. 110	
$\mathcal{I}_V$	Validation set of configurations, p. 110	
$\mathcal{Q}$	Set of states, p. 8	
Φ	Fitness function p. 110	
$\rho(\boldsymbol{x})$	Configuration density, p. 9	
$\sigma(\boldsymbol{x})$	Configuration sum, p. 8	