International Academy of Noosphere
Tallinn Research Group

Cellular Automata, Mainframes, Maple, Mathematica and Computer Science in Tallinn Research Group

Victor Aladjev

Kindle Press – 2022

DEDICATION

The present book is dedicated to my wife ***Galina Aladjeva,*** daughter ***Svetlana Veeroja,*** and grandchildren ***Arthur Veeroja*** and ***Kristo Veeroja.***

ACKNOWLEDGMENTS

I would like to express my deep acknowledgment to the researchers such as: ***Toivo Orav, Vyacheslav Vaganov, Valery Haritonov, Kalle Solba, Ülo Hunt, Rein Veetõusme, Elmot Liiv*** *(Estonia)*, ***Oleg Osipov, Eduard Evreinov, Ratmir Poluektov, Arkady Ursul, Evgeny Scherbakov*** *(Russia)*, ***Oleg Vyuzhin*** *(USA)*, ***Jaroslav Martynenko, Nikolay Gershhorn, Dmitry Grinn, Vladimir Kryzhanovsky, Vladimir Tupalo*** *(Ukraine)*, ***Alexander Nikitin, Michael Shishakov, Valery Boiko, Joseph Klimovich, Evgeny Rovba*** *(Belarus)*, ***Marjonas Bogdevicius*** *(Lithuania)*, and a whole range of others who have made a quite definite contribution to the development of the problematics which is being discussed in the present book.

ISBN: 9798447660208

CONTENTS

INTRODUCTION

The formation of *Tallinn Research Group (**TRG**)* dates back to *1969*, when, as a result of acquaintance with the *Russian* translation of the *R. Bellman* collection [1], we got acquainted with the excellent works of *E.F. Moore* and *S. Ulam* contained in it, as well as with the *J. Myhill`s* work [2], which were played a great role in the emergence of our interest in the problems of *cellular automata (**CA**s)*. In the ***CA**s* theory, *J. Myhill* is known for proving *(along with E. Moore)* the *Garden of Eden theorem*, stating that a cellular automaton has a configuration with no predecessor if and only if it has two different asymptotic configurations which evolve to the same configuration. In the period *1969–1972*, within the Academy of Sciences of the *Estonian SSR*, our research interest was focused on the ***CA**s* problems exclusively in both theoretical and applied aspects. It was during this period that our research on ***CA*** problems was most active and was our main focus. Their results are reflected in the monograph [3], that is the first monographic work in the *USSR* in this direction and was noted as one of the best works of the Academy of Sciences of the *ESSR* in *1972* with the award of a monetary prize; in *1977* the book was noted in Soviet Mathematical Encyclopedia and in Encyclopaedia of physical science and technology [180,181]. We introduced *Russian*–language terminology of the main concepts of the ***CA**s* theory, the main definitions and concepts, however, ***CA**s* itself were defined as "*homogeneous structures"*. These moments became generally accepted in the *USSR*, and then in republics of the former *USSR*.

Subsequently, a special structural unit was formed within the framework of ***TRG*** with a focus on noosphere issues. Or more precisely, the Baltic Branch of International Academy of Noosphere is formed on the basis of the *Tallinn Research Group* and is managed by Prof. *Victor Aladjev*. The *Baltic Branch* is non–profit scientific organization registered by law of Republic of *Estonia*. The following fundamental directions of scientific activity were determined for the *Baltic Branch:*

– mathematical theory of homogeneous structures (cellular automata – ***CA*** *or* ***CA**s depending on context) and its applied aspects;*
– computer science and modern information technologies;
– biomedical researches in the context of the noosphere problematics;
– physical and technical researches in the context of the noosphere problematics;
– infodynamical models of systems in the context of the noosphere problematics (models of systems in infosociety);
– preparation and publication of books and periodicals in these areas;
– University courses and seminars on the above problematics;

– computer mathematics systems and creation of software tools fot them.

From the very outset of our researches on the ***CA****s* problems, first of all, with application accent onto mathematical developmental biology the informal ***TRG*** consisting of the researchers of a few leading scientific centres of the former *USSR* has gradually formed. At that, the ***TRG*** staffs was not strictly permanent and was being changed in rather broad bounds depending on the researched problems. Members of the ***TRG*** participated in work of a lot of scientific conferences and other forums in *Germany, Japan, the USA, Great Britain, Holland, Hungary*, and other countries. The main scientific results of ***TRG*** were published in a lot of periodicals, transactions and proceedings in the *USA, Germany, Great Britain, USSR, Holland, Hungary, Czechoslovakia, Japan, Estonia, Russia, Lithuania, Ukraine, Belarus, Moldova* and other countries. The long ***TRG*** activities in the *Homogeneous Structures* issues received international recognition. At present, the ***TRG*** is as collective member of the *Baltic Branch* of the International Academy of Noosphere and the *IFIP* Working Group in parallel processing and computing by *Cellular Automata* models.

Meanwhile, after above–mentioned period of research on ***CA****s* problems in the future, trends in ***TRG*** interests changed quite often and lay within quite wide limits: ***CA****s* theory, computer science, programming, computer mathematics systems, statistics, automated control systems and others. Moreover, during significant intervals our researches in the ***CA****s* theory were not conducted or were carried out nominally because of the more high priority at that time of other topics.

Our scientific reports [4,5] at the substantial level have represented the reviews of the basic results received by the ***TRG*** on the ***CA*** problems and other scientific and practical activity. Ibidem, the analysis of the ***TRG*** activity instructive to a certain degree for research of the dynamics of the development of the ***CA****s* problems as an independent scientific direction as a whole had been represented. The references list in this book contain quite a few links, whereas a rather complete list of them can be found on the following web–links, namely:

http://www.hs-ca.narod.ru *or* https://ca-hs.weebly.com
https://files.portalus.ru/dl/files/TRG.html
https://bbian.webs.com/publications.htm
https://files.portalus.ru/dl/files/Our_publications_2019.pdf

TRG, originally created under the subject of *Homogeneous Structures (Cellular Automata)*, throughout its creative activity both theoretically and practically worked on a very wide range of issues, including applied technological issues. At the same time, the main orientation quite often and, sometimes, during a fairly long time changed to such directions as programming, computer mathematics systems, computer science, books publishing, automated control systems, statistics and many others.

Our researches presented below *(along with activity within the thematic focus of one or the other organization)* were carried out during stay in such organizations as the Republican Computing Center of the *CSO* of the *ESSR*, the Academy of Sciences of the *ESSR*, the All–Union State Project–Technological Institute of the *CSO* of the *USSR (the Estonian branch)*, *Estonian* branch of the Central Project–Technology Institute of the All–Union Association Soyuztorgsystem, Computing Center of the *Estonian* Republican Office of the State Bank of the *USSR*, Production Association Silikaat Ltd., Project–Technology Institute of Industry of the Ministry of Construction of the *ESSR*, *VASCO Ltd., FIDO Ltd., Sinfex AS, Salcombe Eesti AS*, The International Academy of Noosphere *(the Estonian branch)*. Being in these organizations, along with the scientific research described below, we were engaged in purely applied activities: the training of system and application programmers, the development of industry and republican automated control systems, the introduction of promising computing tools and their software, development of different application software systems. A number of our activities in this direction have been awarded at the departmental, republican and all–Union levels.

Already in the International Academy of Noosphere (*the Baltic Branch)* headed by academician Prof. *V.Z. Aladjev* international collective of the scientists and researchers for the period *1995 – 2022* have preformed and have issued, along with a lot of the journal publications, number of text–books, books and monographs on the modern computer technologies, the probability theory, mathematical and general statistics, mathematical theory of the *Homogeneous Structures (Cellular Automata)*, bio–medical researches, and also total reports of scientific and practical activity of the Academy for the accountable period. Main our publications for the above period are shown in the above WWW sites. Our publications were done in the following countries: *USSR, Russia, Japan, Estonia, USA, Ukraine, Belarus, Germany, GDR, the Netherlands, Lithuania, Hungary, United Kingdom, Czechoslovakia* and some others. This book briefly describes our most active researches and practical developments. I hope, the results presented below will help to clear up both theoretical and applied aspects of the ***TRG*** activity, and inform about achievements in the above areas of the modern science and engineering. Of the presented material, a certain comparative aspect can be rather clearly traced regarding development of the same directions in the *West*, primarily in such fields as mainframes, personal computers, computer mathematics systems and some others. In our opinion, the history of the past often allows to better understand the present and predict the future.

CHAPTER 1: Mainframes and Personal Computers

The chapter presents our developments in the field of programming of various kinds of applied projects both for mainframes and for personal computers. At the same time, it should be borne in mind that work in this field was often interspersed with our study on cellular automata theory *(homogeneous structures)*, focusing on one or another aspect, depending on the emerging at that time circumstances.

1.1. System and applied programming on mainframes

In the mid–*1960s*, a number of rather significant problems in the field of computer technology emerged in the *USSR*. Because of this, there is a need for a quick transition to the mass production of unified computers equipped with a large number of standardized software and peripheral equipment. To solve this problem, it was decided to develop an *UCS (Unified Computers System)* – a *Soviet* series of computers, analogues of the System/360 and System/370 series from IBM, produced in the *USA* since *1964*. *UCS* software and hardware *(only at the interface level of the external devices)* were compatible with their *American* prototypes. In the *USSR*, *UCS* has been actively exploited since *1971*. Meanwhile, in the *ESSR (Estonia)*, the appearance of the first *UCS* required mass training of specialists *(programmers and engineers)* for its operation. Moreover, there was an acute shortage of quality literature for the users. Therefore, along with mastering the *UCS*, in a number of organizations in *Tallinn* we organized appropriate courses for the users, prepared and published appropriate manuals, which received positive responses in the *USSR* as a whole [6,7]. This approach made it possible, within a fairly reasonable time to introduce *UCS* in a number of leading organizations of the *ESSR*, in particular, the Central Statistical Office of the *ESSR*.

Given the prospects for the development of *UCS* models, we decided to focus on the operating system of the *OS (analog of OS/360)*, instead of the *DOS* disk operating system *(analog of DOS/360)*, that was delivered with junior *UCS* models with limited hardware resources. Therefore, in order to ensure the possibility of using *UCS* with *OS* to program the problems of automated control systems *(ACS)*, in particular, *ACS* of trade by us in *1976*, the *MINIOS* operating system was created – an optimized version of *OS IBM/360* for junior *UCS* models [8]. *MINIOS* was at one time quite widespread in *Estonia*, *Russia* and *Ukraine* in the development of *ACS* for various purposes in various institutes and enterprises.

The Database Management System *(DBMS)* is a certain set of software and language tools that allow to create databases *(DB)* and manage data.

DBMS play an important enough role in information processing software systems, in particular in various types of *ACS*. As a DBMS, the DBMS *OKA (an analogue of the IMS DBMS of IBM)* was determined during the development of the Collective Use Computing Center *(CUCC)* of the *CSO* of the *ESSR*. Meanwhile, created as a new technology for the IBM System/360 platform, the system for its effective functioning required more powerful *UCS* models than available at that time in *ESSR*. In this regard, the DBMS was created on the basis of the *MINIOS* operating system and the *OKA* DBMS *(optimized version of the IMS DBMS)*, that have been developed by the author during his stay in the Estonian branch of *VGPTI CSO USSR* [9]. The created systems allowed them to be used on *UCS* with minimal hardware resources, allowing you to significantly expand the applicability of these systems at that time and allowed you to significantly speed up the programming and debugging of software for the *CUCC* users and *ACS* tasks. In order to expand the programming capabilities on computers of the *second* and even *third* generation, we have proposed the method of so–called *"disk transits"* which allowed us to program a number of problems more efficiently [10,113].

In the *CSO* system of the *USSR*, in the tenth five–year plan, the first stages were to service information and computational work not only of statistical bodies, but also of enterprises and organizations of various departmental of four *CUCCs* were created in *Minsk, Tallinn, Tula* and *Tomsk*, the tasks of which provide modes of time sharing, teleprocessing and dialogue with computers, ensuring the functioning of territorial *ACS*. It was at all the main stages of the ***1st*** stage of *CUCC* creation in Tallinn that the ***TRG*** took a rather active part, which was repeatedly noted in the systems of both the *CSO* of the *ESSR* and the *CSO* of the *USSR*. Note, members of our group took a rather active part both in the creation of the *CUCC* itself, and in setting and programming the tasks of the *ACS* of a number of enterprises-users of the *CUCC*, in particular, the *ACS* Trade of the *ESSR*. ***TRG*** also took a certain part in the design of the republican *ACS* of the *ESSR* along with automated system of state statistics.

As part of the design and creation of the *CUCC* of *CSO* of the *ESSR* and a number of other *CUCC* in the *USSR*, we carried out both theoretical and practical developments on parallel information processing systems on computer networks. In particular, the collections [10,11] contain our works on parallel information processing systems, parallel algorithms, their modeling in homogeneous structures, which are the theoretical basis of parallel computing. In particular, one of the purely applied tasks of paralleling was the introduction of parallel processing technology for accounting tasks [8] as well as some other tasks of users of the *CUCC* of *CSO* of the *ESSR*, components of automated control systems.

Some perspectives of further development of the homogeneous structures theory as a formal apparatus of investigation of parallel computational technique are considered. In addition, a formal description of a parallel processing system for *UCS (analogous IBM/360)* in terms of system of algorithmic algebras was given [12]. Our approach to creation of *HPC* on basis of hardware of *CUCC* also presented a certain interest. A working layout of parallel *Assembly* language information processing system for series *UCS/IBM 360/370* was created [13], however the difficult enough situation before the collapse of the *USSR* did not allow this work to be completed. It should be noted that when creating the system, parallel algorithms were used, which were borrowed from some applied control algorithms used in some high–parallel models implemented in control models in homogeneous structures *(cellular automata)*. Along with these directions, we took a very active part in all stages of creation of *CUCC* of the *CSO* of the *ESSR* and its users at both theoretical and the applied level, some aspects of which are reflected in [14-22,113].

The development of the above system software and active participation in creation of *CUCC* of the *CSO* of the *ESSR* repeatedly awarded by the *USSR* Ministry of Trade, the *USSR CSO* and the Council of Ministers of the *USSR* during *1976–1982,* including cash rewards for the successful completion of work on the establishment and commissioning of the first stages of the *CUCC* in the *USSR.*

1.2. System and applied programming on personal computers

To solve the problem of automation of chemical and biological research, in *1974* the Institute of Electrochemistry of the Academy of Sciences of the *USSR* and *VNIKI* systems with numerical software control by the *Leningrad* Electro–mechanical Plant began work on the creation of a programmable keyboard desktop computer for automating the workplace of an experimental researcher, supported by the comprehensive target programs of the State Committee for Science and Technology, the *USSR* State Planning Commission and the *USSR* Academy of Sciences. It is important to note that these programs included not only the development of technical tools for the automation of scientific research, but also the creation of standard automated workplaces based on them. The programs taken formed the basis for creation of the first *Soviet* personal computers.

At the very beginning *80s*, the first serious domestic *personal computers (PC)* appeared in the *USSR*, and in the *mid–80s* we began work of system and applied nature for one of the first Soviet PC *ISKRA–226 (analogue of the Wang 2200)* which were one of the most massive ones at that time. PC *ISKRA–226* was focused on conducting online operational planning calculations, working with local databases as part of information and search engines, solving scientific and technical, engineering, statistical and optimization tasks in online mode. PCs can be effectively used in computer networks as an intelligent terminal or elementary computer at the lowest level of network processing, and also in the various *automated workplaces (AWS)*, in particular, workplace of statistician.

Despite the positive experience in the *USSR* in preparing documentation for software support of computers, technical documentation for *ISKRA–226* left much to be desired. That is why we first of all in [23-26] tried to lay the bridge that would connect the desires and skills of the user with the capabilities of the PC *ISKRA–226*. At that time, these materials were in great demand in the *USSR*, quite understandable representing both the hardware and the software of *ISKRA–226*. Note that these materials were also related to modifications of *ISKRA–226* and software and hardware complexes based on it. In particular, [23-26] describes the technical and software tools of the *ISKRA–226* PC, describes the operators of the *Basic* language and the features of their implementation. Application programs used in the operating environment of *Basic* are given, including also our *A–BASIC* program, which allows to expand its expressive facilities quite significantly. These books were intended for wide range of readers and specialists desires to use the facilities of software–controlled computers in their pfofessional activities. Some books were thereafter republished.

In order to enhance the capabilities of expressive means of the language *Basic* of PC *ISKRA–226*, a rather simple approach was proposed using only exclusively the *Basic* language itself, whose essence is to provide the user with ***17*** new, additional operators of the *Basic* language. In order to be able to work with additional operators in the *Basic* environment, a special program *"INTERPRETER"* has been developed, which allows any source program in the language *Basic* of *ISKRA–226* and containing additional operators, to be translated into an equivalent program in the language *Basic* of *ISKRA–226*. The interpreter algorithm with the source code is presented in [24], allowing to rather easily expand the interpreter on new operators of the language *Basic* of PC *ISKRA–226*.

A number of application software were created of which the *"Metrolog"* package to provide a metrological service for enterprises [24] of dynamic organization, using a data management system with fast access in the mode of direct addressing of sectors of a flexible disk. In particular, the package provides the measuring of liquids and gases expense by method of standard diaphragms [28]. The *"Metrolog"* package was successfully implemented in *1988* at the *"Silikaat"* production association *(Tallinn)* with awarding of the package developers by the Ministry of Construction

Materials Industry of the *ESSR*. At the same time, a number of useful tools for PC *ISKRA–226* have been developed and presented in [26] with source codes, allowing you to easily modify programs to suit the specific user requirements and expanding the scope of their applicability.

With the introduction of the *ISKRA–1030* series of PCs in *1988*, which were significantly more advanced than *ISKRA–226*, we switched our attention to this type of *Soviet* PCs. *ISKRA–1030* is compatible with IBM PC/XT PC based on the *KR1810VM86* processor *(similar to Intel 8086)*. *ADOS (compatible with MS–DOS 2.x), MS–DOS, M86 (CP/M), INMOS (UNIX)* were used as operating systems. In addition, *ADOS* was obtained from *MC–DOS 2.x* by translating the interface into *Russian* in the basic encoding. First of all, as a result of the mastering of this type of PC, we have prepared an extended reference manual describing the architecture and software of PC *ISKRA–1030* in sufficient detail. The characteristics of the main components of the PC and its software, the programming language *Basic*, the text editor and a number of other software tools of a wide purpose are presented in the book [27]. The chapters of the book characterize its content and purpose quite transparently, namely:

1. Architecture of PC *ISKRA–1030*
2. *ISKRA–1030* software architecture and functions
3. Operating system *ADOS* of PC *ISKRA–1030*
4. Basics of programming in language *Basic* of *ISKRA–1030*
5. Useful examples of *Basic*–programs for PC *ISKRA–1030*
6. Expansion of the expressive means of the language *Basic ISKRA–1030*
7. Description and operation of *Parcella* software package
8. *MINIDOS* minimum basis system
9. Personal computer software

Along with our programs that perform important mass procedures, the book presents *MINIDOS* minimum basic system for PC and the *Parcella* package with source code to expand the facilities of the *Basic* language, developed at the Design and Technology Institute of Industry of Gosstroi *ESSR* in *1990*. While *MINIDOS* is an optimized by us version of the basic *ADOS* system, ensuring the efficient operation of PC *ISKRA–1030* on minimal resources. At the same time, we paid great attention to both the experience of working with PC *ISKRA–1030* and the features of its application, as well as the useful recommendations to the user. Whenever possible, useful software tools of both the special and the mass nature, along with the most effective technologies for using these tools to solve some or other applied user applications, were offered.

Since this PC model is unified with such a well–known system as IBM *PC/XT*, this book has become quite relevant for any user operating PCs compatible with the IBM *PC/XT/AT* series. Thus, the book presents both application and system software developed by us for PC *ISKRA–1030* in *1988–1991*. In the future, many of our software designed for PC *ISKRA–1030* have been adapted for more powerful PCs running *MS Windows 3.1* and above. The book was designed for a rather wide range of specialists using PCs in their activities, as well as students and graduate students studying the course *"Fundamentals of computer science and computer technology",* and at one time was quite popular in the *USSR*.

1.3. Special courses and books edition on computer science

In addition to this, along with the development of application and system software for mainframes and PCs, accompanied, as necessary, by the preparation and publication of relevant manuals and books, we published the reference books and special book publications related to some aspects of the computer technology used by us. In addition, in order to provide universities with educational material, we have prepared and published books on computer science, programming languages, statistics and some other university disciplines.

So, book [29] describes the basics of working with mass service software *(MSS) – utilities*. More than *160* PCs utilities are described *(including a number of our tools)* compatible with IBM *PC/XT/AT* and *PS/2*, which are designed to implement such important procedures as virus detection and their neutralization; maintenance of disk file structures expansion of capabilities of monitor, keyboard, printers and disk devices; information and reference; PC operation administration; file archiving and protection against unauthorized access; computational process control; diagnostics and testing of the main components of the PCs system; intercomputer communication. Many of the utilities described were the best or some of the best tools at that time of this type. Given the importance of this type of tools for any PC user, the book appeared at that time undoubtedly was useful for students on the course *"Fundamentals of computer science and computing"* as well as for all PC users.

ChiWriter – a commercial scientific text editor for *MS DOS*, created by *C. Horstmann* in *1986*. It was one of the first *WYSIWYG* editors that could work with scientific texts containing as well as the mathematical and chemical formulae, even on IBM PC *XT* computers that were then common. Our book [30] describes and provides the basics of working with *ChiWriter* editor as well as the basics of statistical analysis on PCs compatible with IBM *PC/XT/AT*. On other hand, our book [31] describes working with Borland firm's *Turbo Pascal*, an integrated development environment for the *Pascal* programming language. *Turbo Pascal* is often used in schools to teach programming, and since the early *1990s*, *Turbo Pascal* has been used in universities to study fundamental programming concepts. The concept, organization and implementation of the *Pascal* version from Borland allowed it to become the standard of the *Pascal* de facto language. The book discusses the structural organization of *Turbo Pascal 5.5* and the purpose of its main components, in it a programming framework with specific examples and useful annexes is presented. And although the book is aimed at readers with little experience with PCs, it can be useful for specialists, as it allows you to not only get acquainted with the *Pascal* language, but also to take a fresh look at the well-known principles of programming. The book is intended for engineers, students and schoolchildren, using PCs compatible with IBM *PC/XT/AT* and *PS/2* in their professional and educational activities.

A characteristic feature of using PCs is the organization of information exchange thru communication channels based on them. It is facilitated not only by the rapid growth of the fleet of different types of PCs, the emergence of affordable technical means, but also by the urgent need to quickly solve a number of important tasks in many applications: various kinds of information services; commercial, exchange and management activities; e–post; banking; records management and much more. In the tutorial [32], the basic principles of the construction and functioning of the both local and global computer networks, as well as the principles of the interaction of network devices, the work of popular network services, such as the World Wide Web and e–post, are discussed network security. The manual is aimed at students studying the discipline *"Informatics"*.

The book [33] is a chrestomathy representing at that time a number of promising and popular software tools, descriptions and basics of working with known packages for personal computers are given, namely: *Sprint* – is a text–based word processor for *MS–DOS*, first published by Borland in *1987; Quattro Pro* – a spreadsheet program developed by Borland, *AutoSketch* – a drawing automation system, *MathCAD* – a package for the verification, validation, documentation and re–use of mathematical calculations in engineering and science, notably mechanical, electrical, and civil engineering, *Expert Choice* – decision–making software that is based on multi–criteria decision making, *NewsMaster* – a rather simple publishing system, *PkWare* file archiving utility and several others. At that, the book concludes with a review article *“Homogeneous Structures: Theoretical and Applied Aspects”* related to the problematics of cellular automata. The chrestomathy was a rather demanded book in the *USSR*.

Having many years of teaching experience in the course *"Fundamentals of Informatics and Computer Engineering",* in the textbooks [34,35] we considered the basics of computer informatics, its application sections that make up their software tools, and methods of working with the most typical of them, as well as the history of the development of computer technologies, its state and development prospects at that time, without which it would be impossible to form a modern computer worldview.

Of particular note is the historical excursion into computing technology and software along with the main formal models of computers, among which the first in the *USSR* presentation at the level of textbooks for universities of a new computational model of highly parallel action – homogeneous structures *(cellular automata)*. The ***1st*** book has survived two editions and was widely in demand in the *USSR*, even today there are many references to it. The both books are intended for students of universities and colleges of natural science, students of courses and all those, who independently master the computer world.

Books [36-39] in *Russian* and *English* contain our lectures on the general statistics theory given for many universities whose programs are focused on economic and non–mathematical profiles. So, book contents [36] is:

Chapter 1. Subject and Method of Statistical Science
1.1. Statistics subject and its location
1.2. A brief tour of the history of statistics
1.3. The principles of organizing the state statistical service
1.4. Objectives of statistics and features of its methodology
1.5. Basic concepts and categories of statistics
Chapter 2. Elements of the Theory of Probability
2.1. The classical concept of probability and combinatorics
2.2. Random variables and laws of their distribution
2.3. Probability distribution characteristics
2.4. Basic laws of probability distribution
2.5. Criteria of basic distributions
Chapter 3. Basics of Statistical Observation
3.1. Statistical observation program and plan
3.2. Basic forms, types and methods of statistical observation
3.3. Statistical observation accuracy issues
3.4. Monitoring the results of statistical observation
3.5. Special issues in reporting and census
Chapter 4. Summary, Grouping, and Presentation of Statistics
4.1. Data summary objectives and content
4.2. Basics of the method of grouping statistical data
4.3. Interval groupings and classifications
4.4. Tabular presentation of statistics
4.5. Statistical distribution series

The chapters cover extensive material on the general theory of statistics - from historical excursion, elementary statistics to elements of probability theory, regression and correlation analyses, analysis of variation and dynamic series, elements of the index method along with discussion of statistical analysis software. We introduced a number of new indices that characterize the creative activity of the researcher. At last, as part of the descriptive statistics software, a number of mass procedures based on the *Maple* system are presented. The existence in the books of a number of non–traditional topics makes their as useful ones for those who somehow deal the statistical analysis of various types in their activities.

CHAPTER 2: Computer Mathematics Systems

Computer mathematics was the result, above all, of solving the problems of classical mathematics through computers. During the existence of computers with limited computing capabilities, we could only talk about a numeric solution of mathematical and engineering problems. But, with the advent of computers of sufficient power and the development of the computer-oriented algebraic computation methods, there were immediate prerequisites for creating *computer mathematics systems (CMS),* which became the main tool of computer mathematics. Today, *CMS* are used to solve various scientific, engineering, educational problems, to visualize data and calculation results, as well as convenient mathematical guides. The development of *CMS* has come many years and today among the most famous universal *CMS* can be noted such as *Maple*, *Mathematica*, *MathCAD* and *Matlab*, among which the leaders are the first two. *CMS* find more and more broad application in a lot of areas both natural and economical and social sciences such as chemistry, mathematics, physics, computer science, technologies, education, etc. Systems such as *Maple, Mathematica, Reduce, MuPAD, Derive, Magma, Axiom, GAP, Maxima, MathPiper,* etc. are increasingly in demand for teaching mathematically oriented disciplines, in scientific research and technology. These systems are the main software for scientists, researchers, teachers and engineers. Research based on *CMS* technology tends to combine algebraic methods well with advanced computational methods. In this sense, *CMS* is an interdisciplinary field between mathematics and computer science, in which researches focuses both on the development of algorithms for algebraic *(symbolic)* and numerical calculations and data processing, and on the creation of programming languages and a software environment for implementing of this type of algorithms for various tasks based on them. Below we have described the results of our quite serious work with *Maple* and *Mathematica* – the undisputed *CMS* leaders for today, which was carried out in the following areas, namely:

– the detailed testing of systems with identification of their shortcomings, limitations and errors;
– lectures on Maple and Mathematica systems at universities in Belarus, the Baltic States, Russia and Ukraine;
– development in the environment of the CMSs as both of system tools that extend and/or improve the functionality of systems, and their various applications in mathematics, physics and technology;
– development of proposals for efficient use of systems and programming in their environment, including hidden capabilities;
– preparing and publishing Maple and Mathematica books and guides.

Our publications [42-73] cover all aspects of working with both *CMS*. The working with systems was carried out by us from *1996* to *2020* with alternating emphasis both between systems and between other types of works. A distinctive feature of the overwhelming number of books is the inclusion of source codes in them that explain certain features of the programming techniques used and also used non–standard programming techniques. The attached to some books are *CDs* with our libraries and software packages. Many of the above books and our software tools for *Maple* and *Mathematica* are Freeware license and available at address https://sites.google.com/view/aladjevbookssoft/home. A number of our books on *Maple* and *Mathematica* in the former *USSR* have repeatedly been recognized as one of the best in computer mathematics systems. Along with the books, we were part of the organizing committees of the international and republican conferences with the presentation of plenary reports on computer mathematics systems containing our developments in this direction [74-91,113,183].

Having published in *1991* year a book on *MathCAD* [40], the first in the *USSR* introducing a domestic reader to the field of mathematical means aimed at automating the solution of mathematical and technical problems on a computer, then books were prepared on systems such as *Mathematica* and *Maple*. Exactly to the last system our attention was directed for a longer time. This was primarily due to the fact that it was this system that was used by me and my colleagues from *Lithuania*, *Latvia* and *Belarus* in a lot of applications of mathematical and engineering–physical nature.

Our publication in *1996–1998* in the *USSR* of books that were among the first books on *Mathematica* and *Maple* systems *(including our numerous lectures on them)* gave rise numerous contacting us with a lot of rather interesting questions on the systems that compete with each other and in many ways are similar. On today, there are more than *500*. The bulk was and is quite trivial, but there were many issues that require quite serious research. None of these questions went unnoticed. So, among this mass of letters, a number of questions were contained, the solution of which initiated the creation of many of the procedures presented in our package and library for the *Mathematica* and *Maple* systems, respectively. They will be discussed below. Taking this opportunity, we express our special gratitude to the authors of the letters, whose questions made it possible to formalize them as separate problems useful both for practical application and for educational purposes. In addition, in the process of studying of these questions, we were able to identify many features, limitations and shortcomings of both systems, along with the discovery of a number of rather interesting them undocumented and hidden possibilities.

2.1. Background of interest in these issues

Our first acquaintance with computer mathematics problems dates back to the mathematical package *MathCAD 2.52* as a new at that time unique means of automating scientific and technical problems of computational nature. *VASCO (**V**ictor **A**ladjev **S**oftware **CO**mpany)*, created in *April 1991*, along with the development of software and research in the field of theory and applications of homogeneous structures *(Cellular Automata)*, set one of its main tasks to prepare and publish a series of books on the personal computers software. In this context, in *May–November 1990*, on a PC *ISKRA–1030* compatible with IBM *PC/XT*, we conducted a rather comprehensive test of the *MathCAD 2.52* package *(hereinafter simply MathCAD)*. The given work was carried out in the framework of creative cooperation with *MathSoft (USA)* with their essential financial support and the opportunity to familiarize themselves with their developments in the field of creating software oriented to solving computing problems in various areas of human activity. As a result of the package operating and comprehensive testing of the package was the book [40], which was the first serious publication of this topic in the *USSR* and was of a reference, methodical and practical in nature. The book was a rather demanded.

The book, along with what is said, is organized in such a way that not only provides all the necessary information on working with *MathCAD 2.52*, but also offers the most effective techniques when working with it, focusing on its strengths and weaknesses. A number of examples given in the book are both illustrative in nature and can be used as the finished software fragments in the practical work of the user. At the same time, the given book presents our *MINIDOS/MathCAD* system, based on the previously developed *MINIDOS* system [27] for PC *ISKRA–1030* and package *MathCAD 2.52* in the process of working with *MathCAD* and focused on PC *ISKRA–1030* with limited resources and providing the following important functions, namely:

– performance improving of the package MathCAD 2.52 on PCs with the limited computing resources;
– ease of use in the package MathCAD 2.52 of the MS DOS instructions.

The created system is rather efficient, reliable and compact, located on ***2 360K*** diskettes. The system was used in many organizations of the *USSR*. All this made the book useful and popular enough material and largely contributed to the growth of interest in package *MathCAD* in the *USSR*.

As part of the subsequent mastering of the *computer mathematics*, we studied the free system of computer algebra *REDUCE*, focused primarily on algebraical solution of physical problems. The system is completely written in language *Portable Standard Lisp* – a dialect of *Lisp*. We have done a rather detailed testing of the first versions of the *REDUCE* system whose results are reflected in the book [41], that presents the description and basis of work in the system with a number of examples and essential recommendations. Subsequently, the *REDUCE* has become close to such well–known systems as *Maple* and *Mathematica*.

2.2. Computer mathematics system *Mathematica*

The beginning of *Mathematica* mastering dates back to *1994*, when the *Mathematica 2.2* was chosen by us to solve and experimental research a number of mathematical tasks, including modelling certain behavioural *(dynamic)* properties of homogeneous structures *(cellular automata)* of dimensions *1* and *2*. The *Mathematica 2.2* was used and tested in Tallinn firms *VASCO Ltd*. and *SALCOMBE Eesti Ltd*. on a *COMPAQ Contura* PC compatible with IBM *PC/AT–486*. In addition, a rather significant point should be noted – the system was tested with a minimum amount of reference information on it, which was limited mainly to its reference information. However, this aspect, along with certain inconveniences, contributed to a deeper and more comprehensive testing. Note that the mastering of the system resulted in two books published in *Belarus* and *Russia* [42,43], which were among the first publications on this topic in the *USSR*. Moreover, *Mathematica* was introduced as one of the chapters in the textbooks for universities [34,35], which helped to familiarize a rather wide range of *Soviet* students with the *Mathematica* system.

At the same time, we began collecting the most interesting procedures that provide useful, quite mass functions, including functions that extend the built–in system tools and/or eliminate certain their shortcomings and limitations, as well as solve specifical applications. Soon our focus was shifted to *Mathematica's* main competitor – *Maple V* [45] – and again we returned to active work with *Mathematica* only with the advent of its ***8^th^*** version. Having resumed active use of *Mathematica* from its ***8^th^*** version in *2010*, we used it with certain intervals until the end of *2020* to solve physical and mathematical problems, model cellular automata and other objects, teach the course *"Computer Mathematics Systems",* preparing publications, testing newly appearing versions, including the last version at that time *12.1.1.0*. The main results of this activity are reflected in our books [66-73,92-94] and in our package ***MathToolBox,*** whose features are briefly presented below. These books are widely used in universities of the former *USSR* in the course *"Computer Mathematics Systems"*, by containing not only courses in *Mathematica*, but also a lot of useful and instructive practical examples of procedural & functional programming, revealing many features and subtleties of programming in *Mathematica* and offering effective methods of programming, organizing user datafiles along with many other aspects of *Mathematica* system use. These books are oriented on a wide circle of the users from computer mathematics systems, researchers, teachers and students of universities for courses of computer science, physics, mathematics, and a number of other natural disciplines. These books will be of interest to the specialists of industry and technology which use the computer mathematics systems in own professional activity. At last, books are useful enough handbooks with fruitful methods on the procedural and functional programming in the *Mathematica* system. Many of these books are included in the lists of mandatory or additional literature on the university courses, magister and postgraduate programs linked with computer mathematics systems and computer mathematics. At last, on the *Internet* you can find a lot of web–sites where a number of our books can be downloaded for free of charge and/or read with or without registration, for example in [113,183].

The package contains the procedures and functions created in process of deep testing, programming of various tasks in *Mathematica* along with preparation of books published in *Belarus, Estonia, Lithuania, Russia, Ukraine* and *USA (https://sites.google.com/view/aladjevbookssoft/home)*. The package contents has been tested in the system *Mathematica 8.0.0 – 12.1.1* on PC *MicroLink 500* with OS *Windows XP Professional (Version 5.1, Build 2600, Service Pack 3)* and on PC *Dell OptiPlex 3020* with OS *Windows 7 Professional (Version 6.1.7601, Build 7601, Service Pack 1) during January 2013 – April 2014* **&** *October 2014 – November 2020*, occasionally, with the considerable pauses in the work. A rather detailed description of software represented in the package along with the most typical examples of its application can be found in the above our books.

The ***MathToolBox*** package contains more than *1420* means of different purpose which eliminate restrictions of a number of standard tools of the *Mathematica* system or complement their alonging with expanding the *Mathematica* software with new tools. In this context, the package can serve as a certain additional effective tool of procedural and functional programming, especially useful in the numerous appendices where some non–standard evaluations have to accompany programming. In addition, tools presented in the given package have a direct relationship to certain principal questions of procedural and functional programming in the *Mathematica* system, not only for the decision of the applied problems, but, first of all, for creation of software extending frequently used tools of the system and/or eliminating their defects or extending the system with new facilities. The software presented in this package contains a lot of useful and effective receptions of programming in the *Mathematica* system, and extends its software that allows to program the problems of various purpose much simply and effectively. The ***MathToolBox*** not only contains a lot of useful procedures and functions, but can serve as a rather useful collection of programming examples using both standard and non–standard techniques of functional and procedural programming in the *Mathematica*. The additional tools composing the ***MathToolBox*** package embrace the next sections of the *Mathematica* system, namely:

– additional tools in interactive mode of the Mathematica system
– additional tools of processing of expressions
– additional tools of processing of symbols and strings
– additional tools of processing of sequences and lists
– additional tools expanding standard built–in functions or the system software as a whole (control structures branching and loop, etc.)
– determination of procedures in the Mathematica software
– determination of the user functions and pure functions
– tools of testing of procedures and functions
– headings of procedures and function
– formal arguments of procedures and functions
– local variables of modules and blocks; means of their processing
– global variables of modules and blocks; means of their processing
– attributes, options and values by default for arguments of the user blocks, functions and modules; additional means of their processing
– useful additional means for processing of procedures and functions
– additional means of the processing of internal Mathematica files
– additional means of the processing of external Mathematica files
– additional tools of the processing of attributes of directories and files
– additional and special means of processing of directories and files
– additional tools of work with packages and contexts ascribed to them
– organization of the user software in the Mathematica system.

The package tools can be successfully used as a fairly good collection of means for programming of mass typical problems in *Mathematica*, that illustrate both standard and non–standard programming techniques in the *Mathematica*. Archive ***Archive76.ZIP*** with this package can be freely downloaded here *(https://yadi.sk/d/2GyQU2pQ3ZETZT)*. The archive contains five files of formats {*nb, mx, cdf, m, txt*}. Such approach allows to satisfy the user using different operating platforms. The memory size demanded for the ***MathToolBox*** in *Mathematica* of version *12.1.1.0 (on platform Windows* ***7*** *Professional)* is a little more *11.72 Mb* whereas the number of tools whose definitions are located in the package is *1424*. Given a rather high level of longevity of basic programming language *Mathematica* which practically unchanged from version to version, the relevance of the package ***MathToolBox*** quite prolonged and the package can be long enough used with subsequent versions of *Mathematica*.

2.3. Computer mathematics system *Maple*

In *1997,* having gained some experience using *Mathematica*, we, on the basis of an agreement with the *MapleSoft* on creative cooperation, began to test the *Maple* system. Under this agreement, we have been provided with system documentation along with all subsequent versions of the system itself. In turn, we carried out a rather comprehensive testing of the system, the development of various projects based on it, the holding of a number of courses in the universities of *Belarus* and the *Baltic States*, the publication on this basis of a series of books and textbooks on *Maple* system and its applications, which were published in *Belarus, Ukraine, Lithuania, Estonia* and the *USA* [95-113]. Along with books, the results of the study system were presented by plenary reports on international conferences on mathematics and computer mathematics systems. At the same time in addition to these books that introduce the domestic user to the *Maple*, as well as to a certain extent advertising it, we familiarized the *MapleSoft* with results of our testing of the system, suggestions and comments; a part of our proposals were taken into account in subsequent versions of the *Maple* system. So, the main results of our activity on the *Maple* problematics are reflected in the above books, reports and in our library ***UserLib6789,*** whose features are briefly presented below. These books are widely used in universities of the former *USSR* in the course *"Computer Mathematics Systems",* by containing not only courses in the *Maple*, but also a number of useful and instructive practical examples of procedural programming, revealing a number of features and subtleties of programming in the *Maple* system and offering effective methods of programming, organizing user data files along with many other aspects of *Maple* system use. These books are oriented on a wide enough circle of the users from computer mathematics systems, researchers, teachers and students of universities for courses of computer science, physics, mathematics, and a lot of other natural disciplines. The books will be of interest also to the specialists of industry and technology that use the computer mathematics systems in own professional activity. At last, the books are useful handbooks with fruitful methods on the procedural programming in the *Maple* system. Many of these books are included in the lists of mandatory literature on the university courses, magister and postgraduate programs linked with computer mathematics systems and computer mathematics. At last, on the Internet you can find a lot of web–sites where a number of our books can be downloaded for free of charge and/or read with or without registration. ***UserLib6789*** was successfully used in the development of a lot of projects of physical and mathematical orientation, sometimes allowing to significantly simplify programming;

library tools are successfully enough used for illustrative purposes when mastering the *Maple* software environment. So, many library means were initiated by conducting a lot of courses on the *Maple* system at different levels, held in *2001–2006* for teachers and doctoral students of a number of universities, as well as researchers from academic institutes of the *CIS*, *Baltic States*, etc. Thus, our activity in using the package, working with letters from readers of our books and conducting a series of courses are three main sources which stimulated the emergence of the ***UserLib6789*** library, attached to most of our books. The additional tools composing the ***UserLib6789*** library embrace all main themes of the *Maple* system. The last version of the library contains tools oriented upon the following kinds of processing:

– *tools of general destination*
– *tools of operation with procedural and modular objects of the Maple*
– *tools of operation with numeric expressions*
– *tools of operation with string and symbolic expressions*
– *tools of operation with the lists, the sets and the tables*
– *tools of supporting of data structures of a special type*
– *tools of supporting of bit–by–bit processing of the information*
– *tools expanding graphic possibilities of the Maple system*
– *tools for expanding and improving the standard means of the Maple*
– *tools of operating with data files and Maple–documents:*
    tools of operating with TEXT and BINARY data files
    tools of operating with Maple data files
    special tools for operating with data files
– *tools of operating with the user libraries*
– *tools for problems solving of mathematical analysis*
– *tools for problems solving of linear algebra:*
    tools of general destination and of work with the rtable–objects
– *tools for supporting of problems of simple statistics:*
    tools of problems solving of descriptive statistics
    tools of problems solving of regression analysis
    testing tools of statistical hypotheses
    tools for analysis of time (dynamic) series.

Basic innovations of the above ***UserLib6789*** library means were rather detailed characterized in our books [95-113,183] and in paper [114].

The ***UserLib6789*** library contains the means created in process of our versatile activity in the *Maple* system during the *1998–2011* periods. The ***UserLib6789*** library contains more than *850* means of different purpose which eliminate restrictions of a number of standard means of the *Maple* system or complement their along with expanding *Maple* software with new tools. In this context, the library can serve as some additional tool of programming in *Maple*, especially useful in the numerous appendices where some non–standard evaluations have to accompany programming. In addition, means presented in the library have a direct relationship to certain principal questions of programming in the *Maple* system, not only for the decision of the applied problems, but, first of all, for creation of software extending frequently used facilities of the system and/or for eliminating their defects or extending the system with new facilities. The software presented in the library contains a number of rather useful and effective receptions of programming in the *Maple* system, and extends its software that allows to program the problems of various purposes much simply and effectively. The ***UserLib6789*** library not only contains a lot of useful procedures and functions, but can serve as a useful collection of programming examples using both standard and non–standard techniques of procedural programming in the *Maple*. The ***UserLib6789*** library has an organization similar to the main *Maple* library, allowing you to work with its tools in the same way as built–in *Maple* tools. So, ***UserLib6789*** not only contains a lot of useful procedures and functions, but can serve as an useful enough collection of programming examples using standard and non–standard techniques of procedural programming in the *Maple*.

The tools composing the ***UserLib6789*** library embrace all main sections of the *Maple* system whiles the library itself with all related data files is located in an archive, intended for *Maple* of versions *6 – 12* on *Windows* platforms 95/98/98SE/ME/NT/XP/2000/2003/Vista/7/8/10. In particular, the archive contains a directory and subdirectories with basic data files of the library for *Maple 6 – 12*, data files general for *Maple 6 – 12*, data file *Maple.hdb* structurally analogous to the data file of the same name of the main *Maple* library *(containing the library help database)*, text data file *ProcUser.txt* containing the source codes of all library tools, instruction on the library installation directly in the *Maple* system and a number of other rather useful materials about the library. The library in the period from *March 2007* to *October 2011* withstood ***7*** versions, of which the latter was supplemented by a number of newly created tools, of which some were included from certain our master classes on programming in *Maple 11*, given in *2010 – 2011* for *Belarus* and *Baltic States*. Many of these library and package tools well complement, sometimes expanding, the standard *Maple* and *Mathematica* tools, respectively.

Having devoted a lot of time to comprehensive work *(mastering, testing, project development, modelling, training, publishing books, etc.)* with *CMS Maple* and *Mathematica (today undeniable CMS leaders)*, we have developed a certain point of view on the comparative aspect of the both systems, which was presented by us in [112,115]. We accumulated many considerations that made it possible to conduct at that time quite definite comparative analysis of both systems. Naturally, that this analysis is to a certain extent subjective, the development of both *CMS* to a certain extent can change preference among them according to one or another indicator, but the presented analysis may be of some interest to users of computer mathematics systems [112,115]. Note, our book [115] is featured in the recommended books lists by *MapleSoft Inc*. and *Wolfram Research Inc*.

In conclusion, let summarize our opinion on the comparative evaluation of the *Maple* and *Mathematica* systems. Both systems are undoubtedly leaders among *CMS* today, but they are replete with numerous errors *(in a number of cases unacceptable for systems of this kind),* the elimination of which is given relatively little attention by developers from *MapleSoft* and *Wolfram Research*. Probably, for commercial reasons, developers often unreasonably release new releases that retain old errors and, in some cases, introduce both new errors and various kinds of the *"architectural"* excesses. This issue has been repeatedly raised both in our publications and directly to developers. However, if *Maple* developers are trying to solve the given problem in some way in open dialogue with users, then *Wolfram Research* takes any criticism *(absolutely justified in the vast majority of cases)* very painfully. A similar list could be continued.

From our experience of rather deep use and testing of both systems, we note that *Maple* is a significantly more friendly and open system, which as a software environment provides a fairly developed built–in *Pascal*–like imperative language of a procedural type, which greatly simplifies the mastery of the package to the user who has experience in modern programming in the environment of one of the procedural languages. Whereas *Mathematica* has to a certain extent *"archaic" (more precisely, rather unusual)* and not so elegant language, in a number of respects, different from many popular programming languages.

Meanwhile, both systems are not universal from the point of view of the programming systems, preventing the user from creating tools that to run outside of the system itself *(that is, their software environment does not fully allow creating* {*exe, com*}*–files with software created in it),* which significantly limits the mobility of the tools created in this way. However, it should be noted here that not everything is so unambiguous, first of all, regarding the built-in language of both systems. Particularly, our analysis of the capabilities of *Maple* and *Mathematica* systems to solve various mathematical problems in scientifically and methodically context was noted in a number of sources, for example, [116]. And to this aspect of using *Maple* and *Mathematica* as a software development environment for mathematical problems solution our books [112,115] were devoted to.

CHAPTER 3: Mathematical Developmental Biology

Mathematical Biology is a section of biology using mathematical models and abstractions of living organisms to study the principles governing the structure, development and behaviour of systems. *Mathematical biology* focuses on the use of a certain mathematical apparatus to study biological systems. *Mathematical biology* is aimed at mathematical representation and modelling of biological processes using mathematical methods and tools. Due to the complexity of living systems, the mathematical biology uses a number of fields of mathematics and has certain contributed to the development of new methods. Attempts to use mathematical methods in biology have been known since the time of *Euler*, but the formation of mathematical biology as a special section of biology occurred only at the beginning of the last century mainly thanks to the works of *Thompson*, *Lotka*, *Volpert*, *Rashevsky* and others. From the same time, the division of mathematical biology into separate independent areas began, such as the mathematical theory of evolution, mathematical genetics, cybernetics, mathematical biophysics, which have achieved sufficient development and recognition to date with a general cognitive objective.

The situation was somewhat different with the mathematical biology of development. Although fundamental works in this field of mathematical biology appeared at the beginning of the ***20^th^*** century *(Schmalhausen, Thompson, Bertalanfi)*, and then they were supported and developed by a number of large biologists *(Waddington, Volpert, Apter and others)* and mathematicians *(Turing, John von Neumann, Tom)*, only the output of a collective monograph [117] allows us to talk about the beginning of the formation of mathematical biology of development as an independent direction in biology. The monograph presented one of the first attempts to bring together various studies related to application of mathematical approaches and methodology in developmental biology, and thereby approved this direction within the framework of mathematical biology. Here we will present some of our results in this direction.

3.1. General prerequisites

The development of organisms, as you known, is a mysterious process. *How can a single cell – a fertilized egg – grow an organism that consists of many millions of cells forming an extremely complex self–regulating system?* The admiration for this process increases even more, if we recall that it is essentially autonomous, that all cells in the body are genetically identical and that development is strictly controlled from the inside. So, speaking of the autonomy of the process, we believe the fact that all the information necessary for the development of the body is contained in the original cell; the external environment provides for the development by certain energy and materials only, not information.

Indeed, a *zygote* of a certain species always turns into an organism of the same species – whatever the environment. *Growth* is carried out mainly through a continuous process of self–reproduction of cells in the body, but *differentiation* of cells in the process of growth is more difficult to understand, since, according to biologists, all cells contain the same set of genetic instructions, i.e. new cells are genotypically identical to their progenitors. In this regard, the question arises: *How do the cells become different from each other and develop into carefully developed spatial forms?* Moreover, the entire development process is strictly controlled so that the different parts of the body develop in certain proportions and in many cases the body is able to overcome sometimes significant damage. Naturally, the development process is based both on rigorous control and adaptation mechanisms. For today, we do not know a better approach to finding out all the issues, except to solve similar problems for suitable artificial systems. In this direction a number of results were obtained.

It should be noted, however, that study of the development phenomenon in the body led some researchers *(Driesch, Elsasser)* to the conclusion that the body cannot be considered as a machine. From the point of view of cybernetics, the general theory of systems and biology itself, it is very important to try to find out the question: *Can a machine even develop like living systems and, if so, how?* This is important to know for two main reasons: *firstly*, if the machine can`t develops, then the argument remains that living systems have certain specific phenomenon. In this case, the argument of cybernetics that living and non–living systems can be quite defined in terms of the same principles and concepts would be called into question. *Secondly*, with a positive response, that is, if the principles of the development of non–living systems were to a large extent understood and a satisfactory analogy with living systems was made, then along with important revolutionary applications in technology of many production processes, we would be able to obtain a satisfactory apparatus for study of living developing systems. For this we used a certain model approach to study a number of important phenomena of developmental biology, based mainly on infinite automata, as which were used cellular automata, that is, a discrete modelling method was used. So, naturally, the discrete approach can`t be seen as an alternative to the continuous development of living systems, but at the conceptual level it may help to clarify some fundamental issues of biological development. Now, below let's consider the main attempts in this direction at that time.

3.2. Discrete modeling in developmental biology

A body development as a rule consists of ***growth*** and ***differentiation.*** *Growth*, as is known, means simply increasing the size of the organism mainly due to the self–reproduction of cells. *Differentiation* is a much more complex process, and it is advisable to distinguish at least ***2*** of its types: *spatial* and *phenotypic*, which *M. Apter* calls *functional*. So, in a growing tissue, a change in the shape and configuration of intercellular communication *(spatial differentiation)* can be distinguished along with an increase in the differentiation of individual cell types *(phenotypic)*. At that, it should be noted that for *spatial differentiation* in biology there is an established term ***"morphogenesis",*** whereas for modelling purposes, in our opinion, the first term is more suitable.

Sure, *phenotypic* differentiation also takes place in spatial differentiation, but for simplicity we consider them separately. The developing organism is characterized not only by the ability to achieve a complex *spatial* and *phenotypic* differentiation, but to a greater or lesser extent it has ability to regulation and regeneration. By ***regulation,*** we mean the property of an organism to develop into a normal individual, even if it was subjected to changes in the process of development *(for example, during the removal or restructuring of cells),* whereas by ***regeneration*** we will understood the property of the organism to restore any disorder that that organism received at the time of its complete development.

Despite the importance of understanding of the *biological development*, including spatial and phenotypic differentiation, regulation, regeneration, and the phenomenon of self–reproduction, attempts to achieve success in modelling this process can be quite attributed to the ***1st*** stage of the *model* period which is characterized by modelling individual phenomena of the development process with a wide enough variety of techniques used for modelling. The principle of research was common to all these models: *the formalization of the phenomenon being studied → building a specific model → a comparative analysis of the functioning of the model and the real biological phenomenon*. The main role of the *first* phase of modeling can be characterized by the fact that a number of complex development processes were given satisfactory formalization that was adjusted based on the analysis of numerous formal models [117-128]. The analyses of a number of models allowed a new look at some regulatory mechanisms of development. Meanwhile, we had a number of models unrelated by the general theoretical base, which complicated to obtain some conclusions.

Naturally, a similar situation did not contribute to the development of a single apparatus for modelling developmental biology. However, within the framework of the *first* stage, two techniques for modelling a number of developmental phenomena were arisen: the *cellular automata* and the *developing parallel Lindermayer grammars*. Cellular automata afterward known as *homogeneous structures (**HS**)* were used by *von Neumann* to study the self–reproduction problem, and parallel developing grammars were first introduced by *A. Lindermayer* to model morphogenesis [129–132] and subsequently were called ***L**–systems*. Homogeneous structures and ***L***–systems were at that time the most common and popular apparatus of cybernetic discrete modelling of development [113,117,127-133]. We define in the most general terms the concept of *homogeneous structures*, at the end of the book they will be discussed in more detail.

A copy of the same finite automaton is placed in each integer point of the ***d***–dimensional *Euclidean* space (Z^d). Each of them is associated with a finite number of neighbouring automata according to a *neighbourhood index (**X**)*, which is the same for all automata of space. At each integer time $t > 0$, an automaton changes its state from the finite set $A = \{0,1,2,..., n–1\}$ depending on the state configuration of itself and all neighbouring automatons at the previous moment $t–1$. At the same time, changes in the states of an automaton are determined by the *local transition function* **σ.** The simultaneous application of the function **σ** to all automata of space defines a global transition function **τ,** which converts one configuration of space Z^d to another. Among all possible states of the automaton of Z^d, the so–called *resting* state (q_o) is distinguished, the essence of which is that the automaton in the state of q_o does not change its state at the next moment if all its neighbours were in the *resting* state. Thus, the q_o state is entered to impose a limit on the *rate* of information transfer to the ***HS.*** So, the ***HS*** is an ordered five $HS \equiv (Z^d, A, X, n, q_o)$; this is the concept of the so–called *classical* ***HS.*** Currently, the mathematical theory of ***HS*** is a fairly well–developed apparatus for the study of many discrete processes *(the **HS** problematics is discussed in more detail at the end of the book)*, which allows by formal means to investigate at the cellular level such developmental phenomena as growth, self–reproduction, differentiation, regulation and regeneration. For today, the ***HS*** have made it possible to implement a number of interesting development models that receive very interesting biological interpretations [117-124,126-128,134]. Along with these problems the ***HS*** can be satisfactorily explored development issues such as complexity of developing systems, processes controlling growth, regulation and regeneration, sustainability of the development processes, necessary and sufficient regulatory and regeneration conditions, etc.

But along with this, ***HS*** give rise to difficulties in modelling of a number of biological phenomena in them. The main difficulties are related to the high sensitivity of ***HS*** to the dimensionality of the models and also to the serious restrictions on the possibility of cell division within the simulated developing organism. Given the difficulties of modelling of a number of biological phenomena in ***HS,*** *Lindermayer* [129] introduced the above–mentioned systems *(**L**–systems)*. Within the framework of ***L***–systems for modelling morphogenesis and growing structures, *Lindermayer* proposed branching algorithms [130], while a number of authors [131] introduced graphical generating systems for modelling development and growth. A number of growing algorithms have been implemented on the basis of ***L***–systems, a review of which can be found in *Lindermayer's* excellent work [129]. *Lück G.* and *Lück J.* [135] also used an ***L***–system to explain tissue growth. Over time, a large number of models of both growth and growth as part of the overall development phenomenon appeared on the basis of ***L***–systems. Therefore, it is appropriate to introduce the concept of the ***L***–systems. We will introduce the concept at a meaningful level.

A ***L****–system* is a triple of the form $\boldsymbol{L} \equiv (\boldsymbol{V}, \boldsymbol{\omega_o}, \boldsymbol{R})$, where:

V *— an alphabet of the system;*
$\boldsymbol{\omega_o}$ *— an axiom, non–empty chain of symbols over alphabet* ***V;***
R *— a set of output rules*.

In the standard version of the ***L***–system, the output rules are of the form $\boldsymbol{v \rightarrow \alpha}$, where ***v*** is a symbol of the given alphabet $\boldsymbol{V}$, $\boldsymbol{\alpha \in V^*}$ is a chain of characters *(possibly empty chain)* in the same alphabet. Thus, each rule can be interpreted either as a division of a cell $(|\alpha| > 1)$, its modification $(|\alpha| = 1)$, or as its death $(|\alpha| = 0)$. If there is no more than one output rule for any symbol of an alphabet ***V,*** then ***L***–system is called *deterministic*. Systems having more than one output rule for some symbols of alphabet ***V*** are called *non–deterministic*. Both types of the system are used.

Thus, ***L***–systems substantially expand one–dimensional ***HS*** in the sense of a plurality of generated words. From the point of view of biological adequacy, they receive quite satisfactory interpretations. ***L***–systems are already well established in describing a number of biological processes and now, in all likelihood, represent the most mathematically developed and biologically adequate discrete apparatus for modeling developmental biology. In relation to the apparatus itself, ***L***–systems are more abstract than ***HS,*** if only because they are not rigidly bound to the coordinate system and, in fact, are one of the types of parallel formal grammars that are intensively studied [132]. At that, it should be noted that ***HS*** may be considered as a certain type of parallel grammar [136-138], which is the proper subclass of the class of all ***L***–grammars. Below, we will analyze the ***HS*** and ***L***–systems in more detail for their capabilities for biological modelling problems, which, sometimes, are quite different.

The biological interpretations of ***HS*** base on the following assumptions:

1. *As a biological unit, the cell that has some cellular automaton is most suitable, and all we need to know about it is the dependence of its output on the entrance and its state.*

2. *All cells in the body have the same genotype, that is, the same set of instructions on its functioning.*

3. *The development of the cell system depends rather significantly on the exchange of information between its cells.*

4. *A body itself regulates the most important aspects of its development. In other words, the development is managed internally, not externally*.

Of course, each of these four assumptions is a simplification of the real state of things, but when models based on them help to achieve a certain clarity, new assumptions can be included in them to bring these models closer to reality. So, in modelling regulation, differentiation, regeneration in ***HS,*** in addition to these assumptions, we used the *Sager* principle on the formation of forms in accordance with the instruction system. This idea of a system of instructions is most attractive precisely because it can serve as a development of that path along which the application of theory of information in developmental biology is usually thought of.

It is easy to verify that the behaviour of the finite ***HS*** can be described in the logical network language. And since *Sugita* proved the possibility of expressing *Jacob–Mono* models in the language of logical networks, than development models in ***HS*** can receive a certain genetic interpretation in the language of *Jacob–Mono* models. The solution to this problem and the results of *Sugita* would then prove the equivalence of *Jacob–Mono* networks and logical networks, from which the fundamental possibility of interpreting models implemented in ***HS*** by *Jacob–Mono* models, will follow. Modelling in ***HS*** allows you to consider development processes from the point of view of hierarchical structures [121-124,127,128]. So, the development models implemented in ***HS*** can be quite investigated by means of systems that are provided by the properties of ***HS.*** This approach allows to obtain qualitatively new results from modelling development processes. To simulate the process of forming axial and multi–dimension structures, that is an integral part of the overall development process, we used several types of ***HS*** [113,119,120,127,128,141,183].

But when discussing modelling the general development problem, it is necessary to add a fifth to the four assumptions mentioned above: *the development of each organism is carried out by self–reproduction of its constituent cells*. It is to these *five* basic assumptions the ***HS*** reciprocate very well. In this case, the cells of the developing organism are answered by ***HS*** automata. In fact, a real cell in its structure is much more complex than our automata, but to study the problem at the cellular level, we must somehow simplify the problems associated with structural complexity of a cell, as well as the functioning of its constituent parts. This is what we achieve, treating the cell as a black box – a *cellular automaton*.

Thus, considering intercellular interactions, we move to more high level of organization than that studied by *Steel* and *Goen* [117], modelling the enzyme systems based on *Turing* machines. So, we take a cell as a unit and in the sense of its behaviour are limited only to the dependence of its output on the entrance and its state at the previous moment of time. In principle, the ***HS*** quite admits ***structural*** level of modelling, when the internal organization of a single automaton of ***HS*** is investigated also when implementing development models in ***HS.*** In some development models mentioned above [128], the internal structures of the ***HS*** single automaton have already been used. Thus, ***HS*** quite allow modelling of developmental processes at lower levels than an individual cell. Time in ***HS*** is supposed to be discrete, whereas in fact it is continuous, but for the purposes of discrete cybernetic modelling of development processes this is not significant yet, naturally with certain essential reservations.

Each cell has the same genotype, on which, apparently, the appearance of the resulting organism depends. Therefore, probably the most convenient way to create effective developmental models is to model the genotype *(cell work program)* [139,140]. Indeed, until now, most of the work on cell differentiation has been carried out at the level of interaction among tissues, while it is extremely significant to extend our understanding to the nature of the processes taking place at the cellular level. With this approach, we throw out single–celled organisms that are experiencing development also, which is mainly the result of intracellular activity. However, at our modelling stage, we are still forced to put up with this. Above, we assumed that all cells have the same genotype, that is, each cell in the body, starting with a *zygote*, begins to work with the same set of genetic instructions. The very concept of the ***HS*** includes exactly this aspect, on which we will dwell a little more.

There is evidence that in a number of organisms different cells may have a different genotype. However, it does not follow from the definition of the ***HS*** itself that all cells of a developing organism simulated on such structures are identical. Indeed, in the presence of the same program of work, the ***HS*** single automata, as a result of differentiation *(changes in internal states)* in different regions of homogeneous space, have different internal states *(phenotypes)* and therefore react differently to the same input signals. Thus, to differentiation of cell phenotypes corresponds in the ***HS*** to differentiation of internal states of single automata of the ***HS.***

In this regard, at each stage of development, all or a certain part of ***HS,*** differentiating, acquires in the general case new properties.

In general, the term *"**differentiation**"* has a number of interpretations. A rather interesting interpretation of this term was given by *M. Apter*. From the point of view of types of differentiation, regionalization is of greatest interest in the first place, or, according to *Apter* [134], the main question is rather to find out how the structure arises first at the organization level *(how cells are "self–marking")* than to establish the nature of physical mechanism implementing this plan, although this mechanism can even lead to an increase in complexity. In this context ***HS*** provide a number of rather acceptable constructive answers to similar questions.

Above, we assumed that the development of the body very significantly depends on the exchange of information between cells. At present, this fact is universally recognized and has a number of evidence in its favour [127,134]. In the case of the ***HS,*** such an exchange of information is the transfer by one elementary automaton to another *(others)* of symbols of the input *(output)* alphabet or a message about its state. In our view, there is no need for a limitation of any kind to the transfer of information from automaton to automaton in the ***HS*** due to the complexity of intercellular interactions *(especially chemical)* in real organisms. Since a body grows from a single cell *(zygote)* by self–reproduction of this cell, than the ***HS*** automaton should have the ability to generate its copy at the right time. This is achieved by the fact that some adjacent non–functioning single automaton of the ***HS*** is transferred to some non–zero internal state, thus becoming functional and an integral part of an already more complex organism. Moreover, it is assumed that once the non–functioning single automaton becomes functional, it will automatically receive the entire genotype *(program of work)* of the original automaton from which the body develops in the ***HS.*** In this we conclude the concise discussion of the properties of ***HS****s* in terms of the simplified biological prerequisites underlying development and proceed to consider the classes of problems relevant to a particular development process which can be modelled and investigated in the ***HS****s* of different types.

The ***first*** class of problems includes the question of how differentiation, regulation and regeneration in the body are carried out. The models built for this purpose in ***HS*** made it possible to clarify a number of issues and formulate interesting problems for further research. Issues related to this class of development problems can lead to a better understanding of the problem of the formation of spatial structure in general. Furthermore, in the process of solving these issues, the ***HS*** concept has been expanded and made more acceptable for biological modelling [119-124,126-128].

The ***second*** class of tasks refers to the study of the growth process that in itself does not constitute a problem from an abstract point of view, since it is provided by the self–reproduction of cells on which the development itself is based. However, this problem involves the question of how the body can limit its size in the process of self–reproduction of cells, if this should be fully provided by the genotype of the cell itself. Indeed, such growth is of interest when spatial differentiation can occur in the process of continuous self–reproduction of the original set of instructions without the influence of external influence. It is also essential to study the growth processes using a limited number of instructions for organisms consisting of a large number of cells. In this regard, it is promising to research the stability of processes of the growth and their controllability in relation to various kinds of disorders, since this may be of some interest for such, in particular, the field as oncology. A certain part of these questions have been investigated through growth models, some of them are discussed somewhat lower.

The ***third*** class of problems – the self-reproduction of organisms. Current models of self-reproduction are characterized by the fact – one organism builds its copy. However, from the point of view of development, it is the question that is of greatest interest: *How can a single automaton of **HS,** having begun the process of self–reproduction, give rise to some complex enough spatially differentiated non–trivial organism that will capable of self–reproduction and to a certain extent to regeneration?* In this regard, the question arises: *How does the process of forming a complex enough spatial organism protects itself from errors and what set of instructions of the original automaton of **HS** can this be achieved**?*** Thus, the range of problems of the *third* class involves an approach to self–reproduction at the level of cells and not organisms.

The ***fourth*** class of tasks can be described as the complexity problem in the biology of development. Here you can formulate a number of very interesting questions about the complexity of a single automaton *(cell)*, from which a complex multi–cellular organism grows, about complexity of spatial differentiation, about the change in complexity in the process of development of the organism, at last about the complexity concept in general. Some of these questions have been explored in growth models and in the research of the ***HS*** in general [113,118-129,126-131,134,139], however the problem is still rather far from our being fully understood.

Each of the listed tasks of the *four* classes described is an integral part of a single development process, but it is now necessary to think about how to decompose development into components and analyze them. At the same time, we should not think that some simple model will reveal all the properties of this development process. Therefore, each of the task classes suggests a number of other possible directions for modelling, and the list of classes itself can subsequently be expanded.

At the initial stage of cybernetic modelling when fundamental possibility itself was in doubt, such an approach using the ***HS*** was perhaps the only possible both in terms of the existence of some suitable apparatus and in terms of the readiness of biology itself. Moreover, ***HS*** themselves largely originated precisely as a tool of such modelling. However, by modelling developmental processes in the ***HS,*** we largely ignored the basis of these processes **–** *cell reproduction*. Indeed, cell division in such models could be carried out only at the boundaries of the body, and its internal cells fundamentally did not have such possibility, since ***HS*** is always tightly tied to the coordinate system in the E^n space therefore the insert of a new cell between the cells encounters insurmountable difficulties. This, in turn, cannot but affect the quality of modelling of development processes in ***HS.*** Therefore, it can already be clearly noted that ***HS*** *(despite the fact that they allow a number of interesting generalizations that significantly expand their capabilities for modelling developmental biology* [117,127, 128,139,140]) will not be able to fulfil the role of the universal modelling apparatus with which would be possible to quite successfully investigate developmental biology as a whole. A similar conclusion can be made and regarding the apparatus of ***L***–systems.

In order to obtain a more acceptable modelling apparatus, *Le Choi* [117] introduced parallel exchange systems which inherit the main features of both systems *(**HS** and **L**–systems)*, although they also are very sensitive to the dimension of space and have poorly formalized elements *(motion of modules in the space)*. Therefore, it is unlikely that a qualitatively new apparatus can be developed on the basis of these formal apparatuses, as was the case with the advent of ***L***-systems inspired by ***1***-dimensional ***HS.*** Based on the detailed analysis of the main shortcomings of the existing simulators we tried to identify possible ways to develop a more adequate simulator of development processes [139,140]. Therefore, this is, *firstly*, the development of special multi-dimensional parallel grammars together with algorithms that allow maximum parallel execution of operations and insertion at any place of the word of any finite subwords; *secondly*, the use of ideas and concepts of the graph–topological apparatus and, *thirdly*, the development of a completely new modelling apparatus which is best adapted for the biology of development [113,144].

*What does it really make sense to focus on now**?*** Above all, about the ***1**st* opportunity. The development of algorithms and grammars that operate with multi–dimensional words is really of great interest. However, here again there are problems associated with dimensionality. Our search for multivariate grammars and algorithms satisfying such properties has not yet led to anything. However, researches in this direction have brought its positive results for modelling development biology [139,140,183].

Moreover, it makes no sense to expect the creation in the near future of a completely new apparatus that is best adapted to the needs of biology of development. As experience has shown, such a fundamental apparatus is not created quickly, and even in an almost empty place. Thus, it remains possible to use the ideas of the graph–topological apparatus. The most suitable approach at present is the discrete graph–topological approach, which really takes into account all these difficulties of previous systems, although it is even more abstract and less evident. Moreover, there are already some models of development using this approach. Among them are rather interesting models of *Apter* [134] and *Lindenmayer* [130]. In conclusion of what has been said I would like to emphasize that we share *Waddington*'s opinion that the basic theory should be to a certain extent similar to the topology of ***n***–dimensional space. Therefore, already now we must outline the most significant phenomena of multi–cellular living developing organisms at the cellular level, look for the most appropriate mathematical apparatus for them, and when describing the development processes, we should not limit ourselves to scope of the usual concepts of cybernetics and mathematics! Below we will represent certain artificial models related to some aspects of the biological development.

3.3. The French Flag Problem

Subject to the above, we inevitably should come to the conclusion, that the differentiation of cells at highly-organized alive bodies is direct result of activity of extremely complex regulator mechanisms. First of all, for us, apparently, the effective enough acting models are really necessary, whose purpose should be to help with formalization of the problem and apparently to discover a key to understanding of the basis approaches to the problem decision at a language of an exact science. In the future the experimental approach to this problem has allowed formulating a number of the concepts interesting and simplifying the problem; among them it is necessary to mark such principles as *dominance* and *gradients* [139,140].

The first rather serious attempt of creation of a working model capable to development and regulation of axial structures was undertaken by *S. Rose* [117]. In further, a lot of interesting enough models has been suggested, whose comprehensive review can be found in [139,140]. However, the most known formal model of differentiation, regulation and regeneration is the *French Flag Problem **(FFP)***, offered by *L. Volpert*. He presented the ***FFP*** during the ***3rd*** Meeting on Theoretical Biology held in *1968* at Villa *Serbellon (Italy)*. The four *Serbelloni* meetings took place in *1966 – 1969* to explore questions related to the theoretical biology. Biologists, physicists and mathematicians raised many questions, identified relevant concepts and tools and stimulated the further developments of theoretical biology. The *French flag* model is a conceptual definition of morphogen. Morphogen is defined as a signalling molecule that acts directly on *(and not through)* cells, causing specific cellular responses depending on the morphogen concentration. During early body development, morphogen gradients generate different cell types in different spatial order. At that, the *French flag* is used to indicate the effect of morphogen on process of cell differentiation, namely: morphogen affects cell states depending on concentration, these states are presented by different colors of the *French flag:* high concentrations activate *"blue",* lower concentrations activate the *"white"* gene, whereas *"red"* is used as the default state in cells below the required concentration threshold. So, the ***FFP*** in its most elementary form is formulated as follows:

*There is a **1**–dimension connected system from **3*m** cells, each of which admits one of the states **"red", "white"** or **"blue";** should be determined the rules of functioning of such cellular system whose final state is the configuration of French flag (**CFF**) which to certain extent is stable to external influences and damages*. For solution of the ***FFP*** in its classical posing a lot of mathematical and automaton models has been offered, and their analysis from biological standpoint has been carried out [117].

In particular, discussions of the ***FFP*** formulation as a formal model of differentiation, regulation and regeneration of axial biological structures for concrete biological objects have been carried out. For the solution and research of the ***FFP*** the models on the base of a few types of cellular automata were used, putting before modelling a lot of tasks. In the first place, a question relative to the minimal complexity of a model that is capable to differentiation, regulation and regeneration interested us.

It is shown, that at modelling of the ***FFP*** even on basis of *polygenic* one-dimensional cellular automata, an algorithm deciding the problem should be an algorithm over alphabet ***A*** whose elements are symbols composing the ***CFF*** [139]. In addition, additional states of the model should admit a reasonable interpretation in the corresponding biological categories. So, the second question is the revealing of those sufficient conditions which would promote a solution of the ***FFP*** along with their rather satisfactory biological interpretation. From this standpoint, a lot of models have been investigated on the basis of the cellular automata concept. In particular, in [118] we presented two of the most interesting models that solve ***FFP.***

The first of these models is able to a rather perfect regulation and slightly resembles the known model of *M. Arbib* however it is more simple and is free from a few defects of his model. Moreover, the basic properties of our model are absence of a gradient and thresholds along with presence in it of polarity, spontaneous self-limiting reactions and a bilateral stream of control information [128]. While the second model based on a cellular automaton uses a special ***P***-automaton with memory as an elementary ***E***-automaton of the model. This model is also capable of perfect regulation and is characterized by ***3*** main properties, namely, *existing of memory*, *polarity* and *spontaneous self–limiting reaction*. A characteristic feature of the model is the absence of a two–way flow of control information. To the first of the marked models *(in terms of the main features inherent in it that determine the **FFP** solution)* is also adjacent a model based on the class of ***1–HS**** structures that allow you to solve ***FFP*** in its generalized formulation. An extension of the ***FFP*** can be determined as follows. In a ***1–CA**** model a finite configuration C_o of length ***r*** of states of elementary automata of the following kind is determined, namely:

$$C_o = \square X_1 X_2 X_3 X_4 \dots X_r \square; \quad X_j \in \{1, 2, 3, \dots, a-1\} \quad (j=1..r)$$

Then, such generalized ***FFP*** is reduced to determination of a functional algorithm of the model *(whose complexity does not depend on a number **r** of elementary automata of a differentiated chain)* allowing to establish and support in the ***1–CA**** model a configuration of structural kind from states mentioned above, namely:

$$C_f = \square b_{1_1} \dots b_{1_q} b_{2_1} \dots b_{2_q} \dots b_{(a-3)_1} \dots b_{(a-3)_q} b_{(a-2)_1} \dots b_{(a-2)_q} b_{(a-1)_1} \dots b_{(a-1)_k} \square$$

$$b_{p_j} = p; \;\; p=1..(a-2); \;\; j=1..q; \;\; q=[r/(a-1)]; \;\; b_{(a-1)_i} = a-1; \;\; i=1..k; \;\; k=r-(a-2)q$$

Because of use for solution of the generalized ***FFP*** of an approach on the basis of cellular automata *(**CA**)* we first of all would like to determine the simplest type of ***CA***–models allowing to solve the above problem. In this direction there is the following result [113,117,139-141], namely:

The generalized FFP determined in a finite alphabet W of general kind cannot be decided by means of an one–dimensional polygenic structure determined in the same states alphabet W.

Hence, for solution of the generalized ***FFP*** even in the class of *polygenic* ***CA***–models we need to use an alphabet, expanded relative to its initial alphabet and, perhaps, along with some other assumptions. So, one of models on the basis of a ***1–CA**** model uses an elementary variant of the symbolical sorting allowing to solve the ***FFP*** in the *L. Volpert`s* staging during no more than ***t = 3*m*** steps; where ***m*** is length of a differentiated chain of automata of the model [141]. In addition, a sorting acts as one of kinds of a logical gradient whereas the model allows making a number of interesting enough conclusions of biological nature. Along with that, the functional algorithm of a ***CA****-model which decides the generalized ***FFP*** allows formulating the following result [113,139-141,183]:

There is a 1–CA* model with alphabet A=*{*0,1,2,3, ..., a–1*}* along with a functional algorithm, whose complexity does not depend on length h of a chain of elementary automata and that decides the generalized FFP during no more than t =* [*h/2*] *steps for sufficiently large values h. A set of all solutions of the generalized FFP which are minimal in temporal attitude is nonrecursive.

The above result is a solution of the generalized ***FFP*** which for today is the best in the time attitude. Of this result, in particular, follows that *for sufficiently great values **a** and/or **h** decision time of the **FFP** approaches asymptotically half of length of a differentiating chain C_o of elementary automata of a **CA***–model*. Consequently, an interesting enough question arises: *Whether exist functional algorithms of any other type that decide this problem for the best time**?*** In our opinion, an essential improvement of decision time of the generalized ***FFP*** defined by the above result not seems possible. In principle, other approaches to this issue are possible.

It is perhaps also worth pointing out that all solutions to the FFP appear to require three basic elements: *(**1**) a mechanism for specifying polarity; (**2**) a mechanism for differential response of the cells, such as thresholds;* and *(**3**) at least one spontaneous self–limiting reaction*. Today, in all the main models that solve ***FFP,*** to one degree or another, these ***3*** conditions or their analogues are traced. This raises the question of determination of a minimum set of conditions for models solving the ***FFP*** as a whole.

A rather interesting question of study of the generalized ***FFP*** for case of the higher dimensionalities arises, when instead of the linear chains the ***d***–dimension $(d \geq 2)$ networks of finite differentiable identical automata are considered. It is shown that, the results of solution of the generalized ***FFP*** rather essentially depend on the kind of ***d***-dimension ***CFF*** too [139-141]. The above ***CA***–models solving the ***FFP****,* in a great extent allow to make clear the questions such as properties of separate automata, nature of connections between them, input/output control impulses, along with a lot of other prerequisites giving rise to dividing of cellular system along axis onto segments, located in a certain order. A rather detailed analysis from biological standpoint of these and other ***CA***-models of regeneration, differentiation and regulation can be found in [113,128,139-141,183]. In particular, we dealt with these issues quite limited and short time.

3.4. The Limited Growth Problem

One of the basic problems of the development – *How can be reproduced a certain organism, using possibly least number of instructions?* That is rather important from the standpoint of understanding of development in alive systems as the *zygote* should be somewhat simpler, than organism itself to which it gives a life. The *second* problem touches the restrictions of the sizes of an organism, growing in various conditions if such process is completely caused by a genotype of cells, self–reproducing during the growth. The *third* range of questions touches study of such growth when a spatial differentiation during continuous self–reproduction of an initial set of instructions without influence of external influence can take place. For answer to that and other questions the various formal models of the growth have been suggested. The current variety of models of growth is explained by prevalence of mechanisms of the restriction of growth of a developing organism that are widely spread as well as itself process of self–reproduction. In addition, research of mechanisms of regulation of growth is urgent for comprehension of the morphogenesis phenomenon since the growth can be considered as one–dimensional analogue of the morphogenesis [121,128,139-141]. The reader can familiarize oneself with the widely enough represented problematics of continuous models of biological growth in collective monograph [117] along with extensive bibliography cited in it.

The certain simplest models of growth were investigated by means of the computer modelling by *S. Ulam* and his colleagues which were among the first initiators of study of the growth phenomenon by the discrete apparatus, however much earlier this problem was being investigated by a number of researchers *(A. Thompson, L. Bertalanfi, etc.)* with use of the continuous apparatus of the modelling. The discrete growth models studied by group of *S. Ulam* are the most suitable for description of some abiotic systems similar to the crystal structures, simple plants or organic molecules, than for real complex biological systems. In spite of that the work with similar models has allowed to clear up a lot of questions of the growth of forms in case of different restrictions *(logical, geometrical and certain others)*. Researches in this direction are quite promising.

At working with discrete growth models of *S. Ulam* we have used the apparatus of classical ***2***–dimensional cellular automata ***(2–CA)***, that has allowed to receive a lot of new rather interesting properties of discrete process of growth that is subjected to various recurrent rules, allowing to study the phenomenon by formal means [141]. The further development of the ***CA*** concept as a basis of discrete modeling of growth phenomenon has been received by *J. Butler* and *S. Ntafos*. In terms of study of growth process the indubitable interest the problem of excitations spread in ***CA**s* models with *refractority* presents. In this direction, we have obtained a lot of quite interesting results [113]. Based on this class of ***CA**s* models a lot of interesting models of excitable environments has been proposed; part of them can be used for research of processes of self–organizing in systems of cellular nature of various type; the more detailed information on this question can be found in works [113,121,128,139-144,183].

Quite interesting problems of optimization arise in connection with the questions of restriction of process of growth. Indeed, the real biological organisms do not grow with no any limits, but completely supervise own growth during all development and vital functions. In this connection *D. Gajski* and *H. Yamada* have investigated the rules of growth in the ***CA**s* models which allow growing forms of the preset limited size [113,121]. The chief task here is reduced to revealing of the greatest possible size of the *passive* configurations generated by classical ***CA**s* models from some simple initial finite configurations. Rather interesting results concerning the lower estimations of sizes of such maximal passive configurations in terms of various key parameters of the ***CA**s* along with rather interesting discussions of biological interpretations of the results received in that direction can be found in [113]. Rather interesting questions of growing of chains of finite automata of a preset length can be found and in rather interesting works [113,139-144,183].

The works marked in this direction enough closely adjoin our results on the *Limited Growth Problem (**LGP**)* considered a few below. The ***LGP*** concerns a class of minimax problems in the ***CA***-problematics, being of a certain interest from the standpoint of developing cellular systems of the various natures. Indeed, the growth process in the real biological systems is limited, is strictly controllable from within, and depends on genetic and of some external factors. Moreover, the ***LGP*** has a certain cognitive significance, allowing estimating in a sense a quantity of the information required for growth of complex multi–cellular organisms. In more detail with the ***LGP*** and interpretation of the received results it is possible to familiarize in [113,121,128,139-144,183].

In view of more applied aspects it is necessary to mark utility of the ***LGP*** for research of questions of information connection of the intercellular interactions of developing systems along with formation of the certain considerations about character of the genetic code. As distinct from the above ***CA***–models researching the ***LGP*** and explaining mechanisms of management by the process of restriction on the basis of the ***CA***–concept, there are a number of other ***CA***–models explaining the phenomenon from certain other standpoints such as similarity principle, thermodynamic laws, adaptation to external environment, mechanic stability, energetic expediency, etc. Diversity of such kind of interpretations is undoubtedly necessary and allows carrying out multifold research of the problems of growth and development as a whole. Thus, it in a certain extent can be considered as a biologic analogue of the principle of complementarity.

Given the technical difficulties that arise during the immersion of quite complex algorithms in classical ***CA***-models, we selected the class of ***CA**** models for the ***LGP*** solution, whose definition we will introduce using the example of a simple one–dimensional case as follows. The structure ***1–CA**** is an ordered four $\langle Z^1, A, I, Fa\rangle$, where the first two components of the object are defined similarly to the case of classical ***1–CA***, ***I*** is a set of control pulses and ***Fa*** is a functional algorithm *(**FA**)* of the structure. The functional algorithm ***Fa*** itself is determined by the set of following discrete equations, namely:

$$a^1(z)_{t+1} = S[I_r, a(z), I_l]_t$$
$$(O_r)_{t+1} = R[I_r, a(z), I_l]_t \qquad a^1(z), a(z) \in A; \quad O_r, O_l, I_r, I_l \in I, \;\; t = 0,1,2,\ldots$$
$$(O_l)_{t+1} = L[I_r, a(z), I_l]_t$$

where $a^1(z)$ and $a(z)$ – the states of single automaton of structure; I_r (O_r) and I_l (O_l) are respectively right and left input *(output)* control pulses of a single ***z***–automaton of the structure. The very essence of functioning in this way of a certain ***1–CA**** is simple and boil down to the following. While in state $a(z)$ and receiving at the input control pulses I_r *(right)* and I_l *(left)* at time $t \geq 0$, at the next time $(t + 1)$ the ***z***–automaton enters state $a^1(z)$ and emits control pulses O_r *(right)* and O_l *(left)* that are determined according to the above equations. At the same time, the output pulses of each ***z***–automaton are input pulses for all its immediate neighbours.

Thus, the set of ***I*** pulses is divided generally into two distinct subsets of output pulses to the left $(0ut_l)$ and output pulses to the right $(0ut_r)$; in this case, relative to the current ***z***–automaton of the structure, it is convenient to conditionally classify output pulses into input pulses *(entering the **z**–automaton from its neighbours;* In^z_l, In^z_r*)* and *output (transmitted by the **z**–automaton to its neighbours;* $0ut^z_l, 0ut^z_r$*)*. Moreover, there are obvious relationships between both types of indices: $0ut^z_r \equiv In^{z+1}_l$, $0ut^z_l \equiv In^{z-1}_r$, $In^z_l \equiv 0ut^{z-1}_r$, $In^z_r \equiv 0ut^{z+1}_l$. Obviously, if the input pulses for a ***z***–cell coincide with the internal states of their respective closest neighbours $(z-1, z+1)$, and the output pulses with its internal state, then the ***1–CA**** and the classical ***1–CA*** with the *Moore's* neighbourhood index are identical and take place the relations $I \equiv A$, $I \cup A = A$. Therefore, ***d–CA**** is a certain equivalent modification of classical ***d–CA,*** which much more adapted to consider a number of applied aspects of the ***CA***–problematics. Note, that a whole series specific application of ***CA**** models confirmed their rather high efficiency, first of all, from an applied point of view [113,144-147].

When using ***d–CA*,*** we are not bound by the limitations that occur in the case of classical structures**;** the method of functioning of ***d–CA**** allows us to focus attention on the essence of the simulated objects themselves, minimizing the additional difficulties of programming the model in ***CA*,*** and the model itself is made significantly more convenient to interpret. It is shown that the ***d–CA**** can be quite successfully used as an acceptable intermediate stage in modelling in classical structures and in studies of a number of questions of their dynamics [144-147]. This approach is based on the fact that any ***d–CA**** can be structurally immersed in the classical structure. In particular, it is shown that: *Any* $1–CA^* \equiv <Z^1, A, I, Fa>$ *are equivalent to classical* $1–CA \equiv <Z^1, A \cup I, \tau, X>$ *with the neighbourhood index* $X = \{-3,-2,-1,0,1,2,3\}$. This result is summarized onto the general ***d***–dimensional case too $(d > 1)$. Moreover, the following result occurs: *Any* $1–CA^* \equiv <Z^1, A, I = 0_l \cup 0_r, Fa>$ *is modelled in strictly real time of classical **1–CA** with the Neumann–Moore neighbourhood index* $X = \{-1, 0,1\}$, *the alphabet* $A^*=A \cup 0_r$ *where* 0_l *and* 0_r *are the sets of output pulses of automata **1–CA*** respectively left and right*. This result is summarized onto the general ***d***–dimensional case too $(d > 1)$ [113]. In any case, it is appropriate to note that classical ***d–CA*** are more preferable for theoretical research of the formal cellular model, whereas ***d–CA**** represent in many respects a more acceptable environment for modelling specific objects, i.e. the both classes of structures represent as if two different sides of the classical cellular model.

In view of the above, consider the so–called the *Limited Growth Problem **(LGP)*** in which is of undeniable epistemological interest from the point of view of developing cellular systems of various natures. In fact that the purely growth of real biological systems is limited, strictly controlled and depends on genetic and a number of external factors. Moreover, ***LGP*** is also of considerable cognitive importance, as it allows us to assess to a certain extent the amount of information required for the rearing of rather complex multi–cellular organisms [113]. Given the technical difficulties which arise when diving quite complex algorithms into classical ***CAs,*** we selected the ***CA**** class of models defined above for the ***LGP*** solution. At the same time, as noted above, any ***d–CA**** is structurally immersed in the classical ***d–CA,*** that allows the final results of the study to be adequately interpreted in the context of the ***CAs*** models. We define the ***LGP*** without breaking the commonality for the class of the simplest structures ***1–CA*.***

Let $\boldsymbol{c_o}$ be a finite configuration of length ***r*** from the states of the single ***z***–automata ***1–CA**** of the following form $\boldsymbol{c_o = sss...sss}$ under $\boldsymbol{|c_o| = r}$. Then the ***LGP*** is reduced to the definition of the functional algorithm ***Fa,*** that allows to grow up from the original configuration a passive *(not changed over time)* configuration of the form $\boldsymbol{c_f = fff...fff}$ of the maximal possible size $\boldsymbol{L = L(c_o, Fa)}$. The best known ***LGP*** solution for today is the next our result [113,140-147,183].

For structures 1–CA* ≡ <Z¹, A, I, Fa> with values #A = 12 and #I = 4m + 17, where m is the possible minimum propagation rate of control pulses in the structure, there is a functional algorithm Fa that allows to grow up passive configurations of the length L of single z–automata in states "f" from the initial final configuration c_o of length r, where the value L will be determined by the following recurrent relations, namely:

$$L = r(2m+1)^{\sum_{j=o}^{n} w_j^2 + 2(2^r+1)}, \quad w_o = 2^{4rm(m+1)}, \quad w_j = 2(L_j - r)$$

$$L_1 = r(2m+1)^{w_o^2+2}, \quad L_j = L_{j-1}(2m+1)^{2^{2L_{j-1}+2}+2}$$

To grow up the final configuration c_f of the specified length L of single z-automata, the functional algorithm Fa requires t=]3/2+1/2m[*L steps of the structure 1–CA*.

Using the introduced concept of structures ***1–CA**** allows to more clearly imagine the very idea of a functional ***Fa***–algorithm of growing, which can be implemented in the classical ***1–CA*** too, however with essentially high costs. So, our idea of a functional algorithm is reduced to recurrent exponential increase in the growth time of the chain of ***z***–automata of the ***1–CA*,*** using a principle of increasing the amplitudes of repeated cycles of passage of the same pairs of control pulses in the structure due to an exponential increase in the lengths of ***z***–automata segments which define the duration of these cycles. A meaningful description of the essence of the implementation of such a functional algorithm ***Fa,*** which solves ***LGP*** in the ***1–CA**** structure, can be found, for example, in [113]. On the basis of the proposed idea, different modifications can be considered that make it possible to significantly improve the above result of the ***LGP*** solution [113,140-147,183]. However, this issue was beyond our attention.

At the same time the growing time of ***z***-automaton chains of the specified fantastic length does not exceed their double length and, when the value of ***m*** increases, than that significantly affects the length of the growing chain, asymptotically tending to the limit of ***t* =]*3/2*L*[. Obviously, that the theoretical limit of the growing time of the chain of ***z***–automata of ***L***–length in the structure ***1–CA**** is ***t*** **=**]***L*/*2***[, but, due to the limited growth and the need to process this condition by a functional ***Fa***–algorithm, this limit is unattainable. At the same time, modification of the ***Fa***–algorithm used for obtaining the above–stated solution of ***LGP*** gives the chance to grow up a chain of ***z***–automatic machines at the same initial prerequisites in time, asymptotically equal ***t*** **=**]***1/2*** **+** ***1/2m***[****L,*** and with length equal to the following value, namely:

$$L = r\,(2m+1)^{\,4^{r+1}+3} - 2m$$

Our analysis of functional algorithms [113,141] that solve ***LGP*** allows us to divide them into ***2*** large classes that are fundamentally different from each other, namely:

*(**1**) algorithms whose essence is to continuously maintain the growth of the figure until a control locking pulse (signal) is obtained;*

*(**2**) algorithms whose essence is to pre–mark the contours of the grown figure and then fill it with some final **F**–symbols (placeholders).*

The functional algorithm underlying the first ***LGP*** solution belongs to the *second* class, while the time–optimal algorithm belongs to the *first* class. Apparently, for growing figures *(configurations)* of the maximal possible size, the functional algorithms of the *second* type are the most acceptable while for growing the figure in the minimal time – the *first* type. In our opinion, the *first* type of algorithms more adequately reflect the essence of growth processes in developmental biology, which is based on both genetic information of the *zygote* as well as the influence of the external development environment. The *first* algorithm used to solve ***LGP,*** based on the propagation in the modelling structure ***1–CA**** of control pulses, is complex enough and in the case of, in particular, any failure will be able to initiate uncontrolled growth of the figure, causing the so-called *cancer* process. Meanwhile, further complication of this functional algorithm Fa allows [113,141] to slightly improve the marginal sizes of configurations grown in the ***CA**** models, and in this regard a rather interesting question arises: *Are there functional algorithms using any other ideas and provide the best results for growing maximal size configurations, all other things being equal conditions**?*** Finally, from the applied aspects of ***LGP,*** it is worth noting its usefulness for the tasks of investigating the information connection of intercellular interactions of developing cell systems, as well as for forming a number of considerations about the nature of the genetic code and the mechanisms of the emergence of various types of carcinogenesis. We analyzed this type of functional growing algorithms compared to the algorithms of growing artificial systems [113,140-147].

CHAPTER 4: Certain Mathematical Problems

TRG conducted research on purely mathematical topics in both pure and applied mathematics: algebra, analysis, differential equations, theory of optimal control, number theory, probability theory and statistics, theory of recursive functions and algorithms, dynamic programming, discrete mathematics, combinatorics, automata theory, etc. Specifically, a number of results were obtained in the mathematical theory of optimal processes, based on the so–called *Pontryagin* maximum principle, that concern the stability of certain optimal differential systems [151]; based on a special type of classical ***2–CA*** models, a number of rather interesting non–trivial properties of the generalized arithmetic *Pascal* triangle and *Fibonacci* numbers are obtained [113]. *S. Ulam* [1] formulated the problem of the existence of a simple universal self–reproducing matrix system whose a *positive* solution would imply the existence of formal reproducible matrix systems. Meanwhile, in paper [152] we proved the absence of a universal reproducible matrix system for a sufficiently large rank. However, for the case of infinite matrices, the problem still remains open. A number of our other results of mathematical research can be found in report [141] and in books presented in [113]. Below we will present only a few of them.

4.1. The *H. Steinhaus* combinatorial problem

Polish mathematician *H. Steinhaus* more than *85* years ago formulated a rather interesting combinatorial problem called *"pluses–minuses"*, whose essence in our terminology comes down to the following [113,141,148]. Let $c(k) = p(1, 1)p(1, 2)p(1, 3) \dots p(1, k)$ will be the first string of binary elements $p(1, j) \in \{0, 1\}$; $(j = 1..k)$. In addition, the values of k are selected only from the set $M = \{3 + 4t, 4 + 4t \mid t = \{0,1,2,3, \dots\}$. Then the elements of the j*–th* string of length $(k - j+1)$ are obtained from the elements of the $(j - 1)$*–th* string of length $(k - j + 2)$ according to a simple recurrent rule:

$$p(j, i) = p(j–1, i) + p(j–1, i+1) + 1 \pmod 2; \qquad (i=1 .. k–j+1;\ j=2..k)$$

It is easy to make sure, this construction results in a triangular figure $T(k)$ consisting of $N = k(k + 1)/2$ characters $\{0,1\}$. Since N – even numbers for values $k \in M$, the following interesting question can be formulated: *Is it possible for any permissible value* $k \in M$ *to determine the figures* $T(k)$*, that will consist of the same number* $m=k(k+1)/4$ *of symbols "**0**" and "**1**"?* In the case of a positive answer, we will say that string $c(k)$ is a solution to the *Steinhaus* problem *(the term "**S–problem**" is later used for brevity)* for a given integer k–value.

A number of professional mathematicians and amateurs were engaged in solving the ***S***–problem, which made it possible to get interesting enough results. Meanwhile, its general decision remained open. And only on the basis of a number of results on classical structures ***2–CA*** together with computer modelling, we managed to get not only a number of new quite interesting results, but also an exhaustive solution of the problem [149]. For further presentation, we will need a number of basic definitions.

Definition 1. The solution S(k) of S–problem for each integer k∈M = {3+4t,4+4t | t=0,1,2,...} will be called a derivative [notation: D(k)] if it is represented in the form of concatenation of the form $D(k) = S(k_1)S(k_2) S(k_3)...S(k_n)$ ***of solutions for*** $k_j < k$ ***values at*** $\sum_j k_j = k$ ***(j=1..n). Let S(k) be a set of all kinds of solutions of the S–problem for a certain value of k. It is easy to verify that S(3) = {000, 011, 110, 101} and S(4) = {1101, 1011, 0011, 1100, 1010, 0101}; these two sets of solutions will be called basis. Derivative solution D(k) is called the basis one [notation: B(k)] if the following defining relations occur in D(k)–representation, namely:*** $S(k_j) \in S(3) \cup S(4)$ ***(j = 1..n).***

The sets of derivatives and basis solutions *(along with their elements)* of the ***S***–problem for each value of ***k*** will be denoted respectively by ***D(k)*** and ***B(k)***. Note, the basic solutions are of particular interest in connection with the fact that they are formed from elementary basic solutions and to a certain extent illustrate one of the interesting examples of phenomenon of the self–complication of quantum character.

To simulate the process of generating said figures ***T(k)***, a special type of classical ***2–CAs*** structures has been defined. A detailed analysis of the dynamics of finite configurations in these structures, basing on studies of the deep properties of their local and global transition functions, made it possible to prove that for each permissible value ***k>2,*** the ***S***–problem has positive ***S(k)*** solutions. Whereas using the corresponding classical ***2–CA*** together with computer simulation, it was possible to obtain some rather interesting properties of the ***S***–problem solutions detailing their structure. The overall result in this direction is as follows [113,141,149,183].

Let S(k), D(k) and B(k) be sets of all, derivatives and basis solutions of the S–problem, respectively, for a certain integer k∈M. Then, for each allowable integer k>2, the set S(k) is not empty, and for each allowable integer k>10, the following relation occurs, namely: #S(k) > #B(k), where #G is the cardinality of a set G.

So, this result gives a complete solution to the ***S***–problem. To research a number of quantitative characteristics of the problem solution we joint used the computer modeling and theoretical analysis of the corresponding classical ***2–CA,*** that allowed to obtain a number of interesting estimates for all types of solutions to the ***S***–problem [113,141-144,183].

For an integer k∈{3+4t, 4+4t | t = 0,1,2,...} the following defining ratios take place, namely:

$$\#S(k) \succ 2^{k-r(k)} \ for \ r(k) \le [k/2]; \quad \#B(k) \ge \begin{cases} 2^{3t-2}, & if \ k \in \{3+4t \mid t = 1,2,3,...\} \\ 2^{3t}, & if \ k \in \{4+4t \mid t = 1,2,3,...\} \end{cases}$$

Similar results occur for derivatives D(k) solutions to the S–problem.

Note that the ***S***–problem can be generalized as follows. Instead of ***A*** from two symbols {***0,1***}, the alphabet ***A*** = {***0,1,2,...,a–1***} typical of ***CA***-models is used, and elements of string ***c(k)*** are selected from alphabet ***A.*** While elements of the ***j***–*th* string of length ***(k–j+1)*** are obtained from elements of the ***(j–1)***–*th* string of length ***(k–j+2)*** according to the recurrent rule:

$$p(j,i) = p(j-1,i) + p(j-1,i+1) + 1 \pmod a; \qquad (i=1..k-j+1;\ j=2..k)$$

As a result, the triangular figure ***T(k)*** is generated from ***N = k(k + 1)/2a*** symbols from alphabet ***A.*** And since the values of ***N*** are integers for an infinite set of values ***k,*** than the following question arises: *Is it possible for each valid* ***k***–*value to define figures* ***T(k)*** *that will contain identical numbers of* ***k(k+1)/2*** *occurrences of symbols from alphabet* ***A?*** In such statement, the ***S***–problem is called *generalized*.

It is reasonable to assume that the generalized ***S***–problem can receive a wider interpretation, namely: *the neighborhood index* ***X*** = {***0,1,2,...,n–1***} *for it is assumed to be arbitrary*. In this setting, the allowable integers ***k*** are selected from the set ***M**** = {***n + t(n–1) | t = 0,1,2,...***}, while the stepped figures ***R(k)*** contain $L = [(n-1)t^2 + (3n-1)t + 2(n + 1)]/2$ of each symbol from the alphabet ***A.*** Under the assumptions made, the *general* ***S***-problem is reduced to the question of having for each permissible ***k***–integer of a ***R(k)***–figure containing an equal number of ***L/a*** occurrences of symbols from the alphabet ***A.*** A generalization of the methods of solving of the classical ***S***–problem allows to formulate the following result [113,141].

For an arbitrary alphabet A = {0,1,2,...,a–1} and an acceptable value k≥2a the generalized S–problem has at least 2a solutions. The number of G(k) solutions for the generalized S–problem at alphabet A = {0,1,2} and admissible values k∈{2+3t, 3+3t | t=1,2,3,...} satisfies the inequality $G(k) > 2^{k-1}$***. For an integer k∈M*, alphabet A and neighbourhood index X = {0,1,2,...,n–1}, the general S–problem has at least 2k solutions.***

Note, that the above results related to the solution of the ***S***–problem can also be generalized to cases of higher dimensions and recurrent rules of a more general form, demonstrating interesting enough examples of self–complication and complex enough reproducibility [113,141,147,183].

4.2. The *S. Ulam* problem from number theory

A heuristic study of the growth problem already in the case of two and three dimensions shows the whole variety of growing figures, which is quite difficult to satisfactorily characterize by formal methods. In view of this, in order to simplify the research of this problem, *S. Ulam* tried to introduce the corresponding definitions in the one–dimensional case with the hope that certain of the basic properties of the so–called *sequences of uniquely defined sums (**SUDS**)* will help clarify the picture in the given direction [1]. However, not so much in terms of the formal problem of growth, but in connection with the number theory, this problem gained fame and attracted the attention of many researchers. The essence of this problem is quite simple and boils down to the following.

On a set $\boldsymbol{M} = \{\boldsymbol{1,2,3, ...}\}$ of positive integers simple binary operation $\boldsymbol{\phi}$ is defined: $\boldsymbol{x + y \Rightarrow z,}$ where $\boldsymbol{x, y, z \in M.}$ The $\boldsymbol{z}$ elements form a set $\boldsymbol{M^* \subset M.}$ The following restrictions are imposed on the given $\boldsymbol{\phi}$–operation:

*(**1**) starting with numbers **a** and **b** (**a<b**), all subsequent elements **z=x+y** are obtained as the sum of any two previous elements **x, y∈M** from the previously obtained sequence, but we do not include those sums that can be obtained in more than one way;*
*(**2**) the numbers themselves do not add up and the most right element of the formed segment (**a, b**) of the **SUDS** must participate in addition*.

The numerical sequence thus obtained will be called ***SUDS(a, b)***. So, the first twelve elements of ***SUDS(1, 2)*** form the following natural numbers, namely: ***1, 2, 3, 4, 6, 8, 11, 13, 16, 18, 26, 28.*** *Twins* in ***SUDS(a, b)*** will be called pairs of adjacent elements that differ in value by $\boldsymbol{p = p(a, b)}$. Below a set of *pairs* of *twins* we will simply denote ***T(p)***. For example, ***T(a + b)*** is a set of twin pairs of the form $\boldsymbol{p(a, b) = a + b.}$ The initial setting of the *S. Ulam* problem consists in determining the cardinality of the set ***T(2)*** for ***SUDS(1, 2)***, i.e. the pairs of adjacent elements of the set ***M**** differing in value by ***2.*** In this regard, *S. Ulam* hypothesized the infinity of the set ***T(2)***. We investigated this problem in a more general statement, for that we will need to introduce a number of additional definitions.

In addition to the ***SUDS(a, b)*** sequence, we will consider the sequence of type ***SUDS1(a, b)***, which differs from the first only in that we not require mandatory participation in the binary $\boldsymbol{\phi}$–operation of the right extreme element of the already formed segment ***(a, b)*** of ***SUDS.*** Note, that both of these ***SUDS*** variants along with self–contained interest in number theory have a number of interesting enough biological interpretations related to the growth problem formalized for the simplest one–dimensional case. In relation to the problem we studied the next questions of ***SUDS*** behaviour:

♦ *definition of partial **SUDS** densities starting from the set element*
♦ *degree of growth of **SUDS** element values starting from the set element*
♦ *changing partial densities of twin pairs relative to the entire **SUDS***
♦ *changing the distance between the nearest pairs of twins in **SUDS***
♦ *estimating the number of pairs of twins in a set **SUDS** segment.*

At the same time, all listed questions relate to both the sequence of the form ***SUDS(a, b)*** and ***SUDS1(a, b)*** for arbitrary integers positive ***a*** and ***b.*** Above all, consider the more complex case of the sequence ***SUDS1(a, b)***. Unfortunately, the algorithm for forming the ***k**–th* element of the set ***M**** *(excluding the natural generation algorithm underlying such sequence definition)* has not yet been discovered. Meanwhile, it has been proved that any ***SUDS1(a, b)*** has an infinite set of twin pairs of at least one of the following types, namely ***T(a)***, ***T(b)*** or ***T(a+b)***. It is shown that if a_k is the ***k**–th* element of ***SUDS1(a, b)***, then the ***k**–th* element of the sequence ***SUDS1(da, db)*** will be the number da_k. This property is also valid for the ***SUDS(a, b)*** type sequences. A completely different picture occurs in the case of ***SUDS(a, b)*** sequences, where we were able to obtain practically comprehensive solutions for a whole series of variants of the generalized *S. Ulam* problem. For example, ***SUDS(1, b)*** for ***b≥5*** has infinite sets ***T(b)*** and ***T(b + 1)*** of twin pairs, and its elements a_k are calculated from simple recurrent formulas, namely:

$$a_k = \begin{cases} b+k-2, & if \quad k \in \{3,4,\dots,b+2\} \\ 4b-2, & if \quad k = b+3 \\ (k-b+1)b+[(k-b-3)/2]-2, & otherwise \end{cases}$$

Density of this sequence relative to the set ***N*** is $\rho = 2/(2b+1)$. ***SUDS(a, b)*** at $a > 1$ and $b/a - [b/a] > 0$ has an infinite set ***T(a)*** of pairs of twins, and its density with respect to the set ***N*** is a value $\rho = 1/a$. The elements of this sequence, starting with $k \geq 3$, are calculated using the simple recurrent formula $a_k = b + (k-2)a$. In works [113,141,146,147], a number of other rather interesting examples of ***SUDS(a,b)*** can be found for which explicit functional relationships of the form $a_k = F(k,a,b)$ can be established along with clarifying of certain other interesting behavioural properties of this type of sequences. Finally, the following result has been proved [141]:

SUDS(1,2) has an infinite set T(2) of twin pairs and its ultimate density relative to the set N is determined by the ratio:

$$\lim_{k\to\infty} \frac{4(2^{k+2}-4)+14}{(12+P_o)(2^{k+2}-4)+P_o*72*5^{2k-10}} = 0$$

This made it possible to get a complete solution to the classical problem of *S. Ulam*. For the study of ***SUDS*** of various types, a special program was developed in the language *PL/1* in *OS/360*, which made it possible additionally to obtain a lot of very interesting empirical results [113].

4.3. An algebraic system for polynomial representation of the a–valued logical functions

The research of a number of classes of *discrete parallel dynamic systems (**DPDS**)* is quite closely related to the study of the properties of ***a**–valued logical functions (**a–VLP**)*. Among the various approaches to the study of such functions, an algebraic approach occupies a special place, when any ***a–VLP*** can be represented by a polynomial *(mod **a**)* of maximal degree ***n(a-1)*** over the field ***A,*** and vice versa, where ***a-VLP*** – a map $R: A^n \rightarrow A$. Meanwhile, in the case of composite number ***a*** far from all ***a–VLP*** can be represented in such a polynomial form, or more precisely *"almost all"* functions do not have such a polynomial representation. And since the alphabet ***A*** in ***DPDS*** can be arbitrary, the problem arises of extending an algebraic method of study to the general case of the ***A***–alphabet. In this regard, an interesting and important problem arises from many points of view: *Is it possible to define an algebraic system (**AS**) that would allow a polynomial representation of **a–VLP** in the **A**–alphabet with a composite **a**–number as in the case of a prime **a**–number?* In this regard, we have defined one type of ***AS,*** in which *"almost all"* ***a–VLP*** have a polynomial representation for the case of the composite ***a***–module [141,150]. Such ***AS*** is defined as follows. A finite alphabet ***A*** = {***0,1,2,...,a–1***} is selected and on it the usual binary addition operation *(mod **a**)* is defined. At the same time on ***A***–alphabet, a binary # multiplication operation is defined according to the following multiplication table. It is easy to make sure, the # multiplication operation on set $A_a \backslash \{0\}$ forms a finite cyclic group $A^{\#}$ of degree $(a-1)$. Relatively to the ***AS*** defined thus, the main result is:

There is an algebraic system <A_a; +; #> in which "almost every" a-VLP defined in the A–alphabet (a – a composite number) can be represented in the form of a polynomial $P_{\#}(n)$ (mod a), where:

#	0	1	2	3	4	5	.	.	a-6	a-5	a-4	a-3	a-2	a-1
0	0	0	0	0	0	0	.	.	0	0	0	0	0	0
1	0	1	2	3	4	5	.	.	a-4	a-3	a-2	a-1	0	a-1
2	0	2	3	4	5	6	.	.	a-3	a-2	a-1	0	a-1	1
3	0	3	4	5	6	7	.	.	a-2	a-1	0	a-1	1	2
4	0	4	5	6	7	8	.	.	a-1	0	a-1	1	2	3
5	0	5	6	7	8	9	.	.	0	a-1	1	2	3	4
6	0	6	7	8	9	10	.	.	a-1	1	2	3	4	5
..	..	..	..	..	..	..	.	.	..	..	..	..	..	..
a-3	0	a-3	a-2	a-1	1	2	.	.	a-9	a-8	a-7	a-6	a-5	a-4
a-2	0	a-2	a-1	1	2	3	.	.	a-8	a-7	a-6	a-5	a-4	a-3
a-1	0	a-1	1	2	3	4	.	.	a-7	a-6	a-5	a-4	a-3	a-2

(+) **– *conventional addition operation*** *(mod **a**)*
(#) **– *multiplication operation defined according to the above table***

$$P_{\#} = \sum_{j=1}^{a^n-1} C_j \# X_1^{d_{j_1}} \# X_2^{d_{j_2}} \# \dots X_n^{d_{j_n}} \quad (mod\ a) \quad - \ a\ polynom$$

which is not containing dyadic expressions of the following form:

$$p_d \# X_j^d + B_d \# X_j^{a-d-1} \qquad (0 \le d_{i_j} \le a-1;\ \sum_{j=1}^{n} d_{i_j} \ge 1;\ X_j, C_j \in A_a)$$

$$p_d + B_d = a;\ \ p_d, B_d \ge 1;\ \ X_j^p = X_j \# X_j \# \dots \# X_j;\ \ \ j = 1..n;\ \ d = 1..[(a-2)/2]$$

This result played a very important role in ***DPDS*** studies for cases of the alphabet ***A=*{*0,1,2,...,a–1*}** ***(a** – a composite integer)* and made it possible to obtain a number of very interesting results concerning the problems of cellular automata, some of which are discussed below. At the same time, the given result gives a completely satisfactory analytical representation of ***a–VLP*** in the case of a composite ***a***–module. For example, even such a very simple logical function as:

$$R_1(x) = \begin{cases} 0, & if \quad x = 0 \\ 2, & if \quad x = 1 \\ 1, & otherwise \end{cases}$$

which is defined in the alphabet A_6, can`t be represented by a polynomial *(mod **6**)*, whereas in ***AS*** $<A_6; +; \#>$ its representation has the following simple form: $R1(x) = P_{\#}(1) = x^2 + x^3$ *(mod **6**)*. A number of other rather interesting examples of this nature, as well as a comparative analysis of the algebraic system determined above along with a classical algebraic system of the form $<A_a; +; x>$, in which operations *(+), (x)* are ordinary binary operations of addition and multiplication by *(mod **a**)*, respectively, can be found in our works [113,141-144,183].

Moreover, based on the algebraic system introduced above, an interesting enough type of classical cellular automata with a sufficiently high degree of reproducibility of finite configurations can be determined. To this end, for ***1***-dimension classical cellular automaton, the local transition function is defined as follows:

$$x_1 x_2 \dots x_n \to x_1^1 = 0, \quad if \quad (\forall k)(x_k = 0)$$

$$x_1 x_2 \dots x_n \to x_1^1 = \prod_{k=1}^{n} \# \delta(x_k), \quad otherwise$$

$$\delta(x) = \begin{cases} x, & if\ x \ne 0 \\ 1, & else \end{cases}; \qquad x_1^1,\ x_k \in A \ \ (k = 1..n)$$

where **#** – an operation is determined according to the multiplication table presented above. It is shown [141], the considered class of ***1***–dimensional cellular automata is characterized by the presence for them of property of essential, but not universal reproducibility of finite configurations.

CHAPTER 5: Cellular Automata *(Homogeneous Structures)*

The closing chapter presents the main fields of research of the *Tallinn Research Group (**TRG**)* on the problems of cellular automata *(**CA**s)* and the results obtained in this direction. It was ***CAs*** problems underpinned the formation of a group of ***CAs*** interested researchers as a ***TRG*** in *1970* after familiarization in *Leningrad (now Saint–Petersburg)* in *1969* with a Russian translation of the excellent collection [1], that contained articles by *E.F. Moore, S. Ulam* and *J. Myhill* that stimulated our research on the ***CA***-problematics. And, if in the *first* few years our main focus was on this area of research, then in the future this area was periodically *(sometimes for a fairly long time)* overlapped with other scientific and technical areas considered in previous chapters of the book. Meanwhile, the ***CAs*** issues, including its applications, primarily in computer science, mathematical and developmental biology, have been a major outline of ***TRG*** activity. At the same time, during the period of activity in certain other areas, for example, computer mathematics systems, we in parallel done computer modelling of a number of tasks of ***CA***-problematics *(along with computer research of purely theoretical problems)*, as well as problems somehow that are related to this problematics *(in particular, problems of biological nature)*. Some parallel algorithms describing certain processes in the ***CAs*** environment were used by us in the development of high–performance parallel architecture computing systems [10,11,113,141-144,175,183].

However, not all of our activities made it possible to actively conduct a research in the ***CAs*** problematics; moreover, we had significant intervals during which we did not conduct any research in the ***CA***-problematics. As a whole, such intervals fall on the period of the collapse of the *USSR* as a single state. In the same periods when work was carried out that was not related to ***CAs*** problematics, attempts were made to find common ground with the current work. In a number of cases, such approach has yielded positive results, allowing to study into certain ***CAs*** problems in parallel. At the same time, such approach made it possible to obtain, sometimes, non–standard solutions for purely applied problems, far, at first glance, from the ***CAs*** problematics [113,141].

As part of a brief historical survey, we will present the main stages of the formation of *the Cellular Automata theory*, including the results obtained by the ***TRG** (1969 – 1998)*, and subsequently by the Baltic Branch of the International Academy of Noosphere *(1999 – 2021)*. Note, the survey is largely based on our research experience in this field since *1969*, i.e. at the dawn of this line of study in the *USSR* and *Estonia*. References cover mainly book publications, while numerous articles are available in [154].

5.1. Basic concepts of classical cellular automata

As the main object of research, the so–called *classical cellular automata (**CA**s)* are considered, which in all their generality are highly formalized models of certain abstract *"Universes"*, developing according to rather simple rules and consisting of simple enough identical elements. ***CA**s* of this type develop according to local *(and everywhere the same)* rules for interaction of the elements forming them. In this context we can consider ***CA*** as a certain analogue of the physical concept *"field"*. The ***CA*** space is a regular lattice whose cell represents a certain identical element *(a finite automaton)* that receives a finite number of states. At that, the history of development of such ***CA*** is set on a discrete time scale ***(t = 0,1,2,3, ...)*** by a finite set of commands changing the state of any elementary automaton at time ***t > 0*** depending on state of oneself and states of its neighbouring automata at the previous time ***(t–1)***. Say that the function acting on each elementary automaton in its neighbourhood is called the *local transition function (**LTF**)*, whereas its action on the entire ***CA*** space determines the so–called *global transition function (**GTF**)*. Change of configuration for such a ***CA*** model under the action of ***GTF*** determines the dynamics of its functioning over time; this aspect plays a major role in the researches of behavioural *(dynamic) **CA**s* properties, including their appendices.

The ***CA**s* can well be considered as the theoretical basis of some artificial systems of parallel information processing or as some kind of acceptable presentation environment for conceptual as well as practical models of spatially distributed dynamical systems. In addition, from a logical point of view, the ***CA**s* models themselves are infinite abstract automata with a specific internal structure that determines a number of rather important properties that quite successfully allow them to be used as a new fairly promising environment for modelling various discrete processes using the maximal parallelization mode. In general, ***CA***–problematics can be considered as a structural–dynamic component of the theory of infinite automata with a certain specific internal organization that is qualitative in nature along with its important enough applied aspects. In general, our point of view on the place of ***CA***–problematics in the modern knowledge system is presented in the last section of this chapter.

Here we will present the basic concepts, definitions and designations that relate to the concept of classical ***CA***–models and used throughout further consideration. A detailed discussion of the basic concepts of ***CA***–models along with the issues related to them will allow a deeper understanding of the foundations of this field of the general theory of infinite abstract automata. Above all, note that the consideration of the material is based on the so–called classical concept of ***d***–dimensional cellular automata

(d–CA, d≥1), regarding which a number of basic definitions and concepts are introduced and some related results, including an important question of the degree of generality of the classical concept too. The definition of an arbitrary classical ***d–CA (d ≥ 1)*** is axiomatically introduced as follows *(henceforth, we will use the signage **CA** for both the individual cellular automaton and their set; the meaning of this designation easily follows from the context and does not cause any misunderstandings).*

The classical d–dimensional cellular automaton (d–CA; d≥1) is defined as an ordered set of the following four components, namely:

$$d\text{–}CA \equiv <Z^d, A, \tau^{(n)}, X>$$

where A is a finite non–empty set (a states alphabet), and this is the set of states that each elementary automaton in the d–CA can take.

The states alphabet ***A*** includes a separate element which is called the ***rest*** state *(indicated by the symbol «**0**»; in addition, for convenience, in some cases, the symbol «**0**» is replaced by the symbol «□»)*. The essence of the special state will be clarified a bit later. Without disturbing commonality, we will use the alphabet $A = \{0,1,2,..., p-1\}$, which contains ***p*** elements **–** integers from ***0*** to ***p–1,*** as its states. Elements of alphabet ***A,*** including the rest state, allow different interpretations in a rather wide range [155].

The component Z^d is a set of all ***d***–dimensional tuples of the coordinates of all points in the Euclidean space Z^d, i.e. Z^d is an integer lattice in Z^d, it serves to spatially identify individual automata of the ***d–CA****s*. It is shown [113] that such lattice does not give anything fundamentally new for the fundamental properties of the dynamics of configurations *(both infinite and finite)* in classical ***d–CA****s*, so for study purposes it is enough to limit itself the integer lattice Z^d**.** The lattice Z^d *(d≥1)* defines the uniform space of ***d–CA*** in which they operate. So, Z^d is a set of all ***d***–dimensional tuples of integers that is used to name ***d–CA*** cells and is called a ***space*** in which all functioning elementary automata are identical.

The element in the Z^d lattice can be considered as the name or address of a particular elementary automaton which occupies this position in the Z^d space. At the same time, it is often convenient to identify an ***j***–automaton located in an ***j***–cell with the ***j***–cell itself. In many applied aspects of ***d-CA (d≥1)*** their geometry plays a rather important role *(therefore, the question of lattice geometry takes on special importance in the structural theory, when the properties of **d–CA**s are considered depending on their internal organization)*, but in our ***CA****s* research this question was considered quite rarely, and then at a purely applied level [113,141,144,175,185].

The dimension ***d*** of the ***CA***–models space plays a fairly significant role, differentiating the entire set of the models into two different subsets: ***1*–**dimensional ***(d = 1)*** and ***d*–**dimensional models ***(d ≥ 2)***. Transition from a ***1*–**dimensional to ***2*–**dimensional case not only dramatically changes the dynamics of ***CA****s* models, which is due to an increase in dimension, but also increases the complexity of most of the problems solved on them. In particular, it is shown [144] that certain dynamics problems for classical ***1–CA*** and ***d–CA (d≥2)*** are solvable and unsolvable, respectively. In most cases, the proof of intractability faces significant difficulties that can be fully attributed to the ***CA*–**problematics as a whole.

Class of ***1–CA****s* represent a special subclass of all ***d–CA****s* ***(d ≥ 1)***, studied quite efficiently. If, in terms of modeling itself, the ***1–CA,*** in our opinion, have no special prospects, nevertheless they are of certain interest as an independent mathematical object. At the same time, using the example of ***1–CA****s*, it is much easier to master the concept of classic ***CA*–**models. So, a lot of types of ***1–CA*** were most intensively researched from theoretical point of view; in addition the vast majority of both theoretical works and their computer modelling for the purpose of research of certain dynamic properties were devoted to this class of ***CA*–**models [164]. A copy of the *Moore* automaton with the alphabet of states ***A*** is placed in each cell of the lattice Z^d *(the output of such automaton is determined by its current state)*. The state of such automaton at time $t > 0$ is a function of its inputs at time ***(t–1);*** at that, output signal of automaton at time $t > 0$ is identical to its internal state. Then each cell of lattice Z^d will determine the name *(coordinate)* of the elementary automaton located at such point. For the sake of convenience, we will identify the points of the Z^d lattice with the elementary automata located in them. So, the two terms *«automaton* ***z****»* and *«automaton with a coordinate* $z \in Z^d$*»* we will assume identical.

We further consider that component ***X,*** called the ***d–CA neighbourhood index,*** is an ordered set of ***n*** elements of the Z^d lattice, which serves to determine the automata adjacent to each elementary automaton of the ***d–CA,*** that is, those automata with which the given automaton is directly connected by information channels. So, in the simplest example ***2–CA*** we quite can imagine a lattice Z^2 in the form of cell paper, where each cell contains a copy of a certain *Moore* automaton. Then X_n={*(0,0), (0,1), (1,0), (0,-1), (-1,0)*} and X_m = {*(i, j)*} (*i, j*∈{*0,1,-1*}) are called *Neumann* and *Moore* neighbourhood indices, respectively. These neighbourhood indices ***X*** have long become classical and are widely used in research of both theoretical and applied aspects of ***d–CA****s*, while the *neighbourhood patterns* ***(NP)*** defined by them have transparent geometric presentation. In general, the neighbourhood index can determine any finite network of elementary automata of the Z^d lattice [3]. Typically, ***CA*** neighbourhood pattern are arbitrary; they take the form defined only by the application aspects of ***d–CA*** models mainly and can be a wide variety of.

Despite the versatility of the simplest neighbourhood indices *(any classic **d–CA** is modelled with a **d–CA,** but with an elementary neighbourhood index)*, more complex neighbourhood indices are used for a number of applications and theoretical research. This approach in many practically important cases makes it possible to significantly simplify the process of embedding specific processes, objects, phenomena and algorithms into classical ***d–CA.*** At the same time, this approach is very effectively used for theoretical studies of ***d–CAs*** $(d \geq 1)$, in particular, in their computer research. Thus, the neighbourhood index of ***d–CAs*** $(d \geq 1)$ is a ***n***–tuple of different ***d***–tuples of integers; it is used to determine the neighbours of a cell, that is, those cells from which the cell directly receives information. Then n neighbours of a certain cell z are cells $z + \alpha_o, z + \alpha_1, ..., z + \alpha_{n-1}$, where $X = \{\alpha_o, \alpha_1, ..., \alpha_{n-1}\}$. If the index X contains a point $0^n = \{0, 0, 0, 0, ..., 0\}$, then each elementary automaton will be in its own ***NP.*** Without limiting commonality, we will generally assume that index X contains a point 0^n that defines the central automaton of ***NP.*** In general, it has been proved that the dynamics of ***d-CAs*** $(d \geq 1)$ do not depend on the choice of an automaton of the ***NP*** as the *central* one. Among all neighbourhood patterns, *coherent* and *incoherent* are distinguished; generally speaking, this parameter significantly affects the dynamics of ***d–CA. NP*** is called *coherent* if the area occupied by it is *coherent* in the topological sense; otherwise, ***NP*** is called *incoherent*. A detailed analysis of both types of ***NP*** in the context of their effect on the dynamics of ***CA***–models can be found in [141]. Next, as a rule, we will deal with coherent ***NP,*** bearing in mind that an arbitrary incoherent ***NP*** can always be replaced by a certain equivalent coherent ***NP*** of the same maximum size, using the appropriate insignificant elements in the ***NP.***

So, the first ***3*** components of arbitrary ***d–CA*** $(d \geq 1)$: the states alphabet ***A,*** space Z^d and neighbourhood index X form a so–called *homogeneous space*. *Homogeneous space* is a static part of ***d–CAs*** $(d \geq 1)$ that describes the physical structure of ***d–CA,*** but it does not define the interactions that will take place between elementary automata in Z^d, i.e., strictly speaking, the above ***3*** components do not determine the dynamics of ***CA***–models.

To determine and study the functioning *(dynamics)* of ***d–CAs*** $(d \geq 1)$, it is necessary to have means to describe the current state of the entire space Z^d at any time $t > 0$. The state of the entire space defines a configuration ***(CF)*** of Z^d, that is, simply the complete set of current states of each unit automaton in Z^d. So, a configuration is an arbitrary mapping $CF: Z^d \rightarrow A$;

let ***C(A, d)*** denote the set of all configurations with respect to Z^d and ***A,*** i.e. $C(A, d) = \{CF|CF: Z^d \to A\}$. The special symbol «□^{d}» denotes the completely zero ***CF;*** $□^d: Z^d \to 0$, i.e. when all elementary automata in Z^d are at rest state ***«0».*** By identifying the states {***«0», «□»***}, we will use the second of them to denote infinite areas of space Z^d filled with automata only in the rest state ***«0».*** The state ***«0»*** has numerous and quite natural interpretations from an applied standpoint. It must be kept in mind that all results given below concerning the rest state ***«0»*** are fair also for the general case of a rest state $h \in A$, i.e. for all classical models ***d–CA*** $(d \geq 1)$ *(moreover, we will understand both a separate configuration, and their set in case of lack of ambiguity as an abbreviation* ***«CF»****).*

The set of all configurations ***C(A,d)*** is heterogeneous relative to the ***d-CA*** dynamics due to the presence of the rest state ***«0»*** in it, so we define two of its main subsets: the *finite* ***CF*** $C(A, d, \phi)$ and the *infinite* ***CF*** $C(A,d,\infty)$. The ***CF*** of classical ***d–CA*** is called *finite* if it contains a finite number of elementary automata in states other than the rest state ***«0»,*** otherwise it is called *infinite*. Obviously, the following relations $C(A,d,\phi) \cap C(A,d,\infty) = \varnothing$ and $C(A,d,\phi) \cup C(A,d,\infty) = C(A,d)$ *(∅ is an empty set)* take place, whereas the ***d*** dimension of the configurations is determined by the dimension of classical ***d–CA*** $(d \geq 1)$. Taking into account the specifics of classical ***d–CAs,*** which is largely due to the presence of the rest state ***«0»,*** along with a number of other rather important reasons, we will henceforth attribute the completely zero configuration c_o to the set $C(A, d, \phi)$. This approach yields many very interesting results regarding the dynamics of the classic ***d-CA*** $(d \geq 1)$. This applies in particular to the problems of reversibility and nonconstructability discussed below. A sufficiently detailed discussion of the above concepts and definitions can be found, for example, in [161].

Operating of ***d–CA*** $(d \geq 1)$ occurs at discrete time $t = 0, 1, 2, 3, \ldots$ and is determined by the *local transition function* ***(LTF)*** $\sigma^{(n)}$, that sets the state of each elementary automaton at the current time $t > 0$ based on the states of its neighbouring automata *(according to the neighbourhood index* ***X****)* at the previous moment $(t–1)$. In other words, the ***LTF*** $\sigma^{(n)}$ is an arbitrary mapping $\sigma^{(n)}: A^n \to A$; below, the following main designations will be used for ***LTF,*** namely:

$$\sigma^{(n)}(a_1, a_2, \ldots, a_n) = a_1^*; \qquad a_j, a_1^* \in A \quad (j = 1..n) \tag{1}$$

$$a_1 a_2 \ldots a_n \Rightarrow a_1^* \text{ – a set of parallel substitutions} \tag{2}$$

where a_j is the state of a ***z***–automaton of the ***d–CA*** and all its neighbors according to the neighborhood index $X = \{x_1, x_2, \ldots, x_n\}$ at the moment $(t–1)$ and a_1^* is the state of the ***z***–automaton at the next moment $t > 0$. A detailed discussion of the arbitrariness of choosing of central automaton of ***NP*** can be found in [141]. Whereas in each particular case, as a rule, for the ***NP*** the most suitable center automaton is selected.

The representation of a ***LTF*** by formula ***(1)*** is most convenient in many respects. In many interesting cases, the approach is useful, and quite real, but, in some cases the use of ***LTF*** in the form of parallel substitutions ***(2)*** is required and is the only possible. The set of parallel substitutions ***(2)*** defines a certain program *(parallel algorithm)* for the functioning of ***CA*** models; parallel replacements ***(2)*** represent a low–level parallel language of programming in the ***CA***–environment. The formula representation of ***LTF*** $\sigma^{(n)}$ is particularly preferred for the computer implementation of ***CA*** while parallel substitutions are indispensable in the programming step of a number of specific ***CA***–models. Questions of the ***LTF*** representation are discussed in sufficient detail in [113,141-144,156-161,182-196].

Basically, we consider ***d–CA*** models whose ***LTF*** $\sigma^{(n)}$ satisfy the defining condition $\sigma^{(n)}(0,0, ..., 0,0) = 0$, i.e. models with limitation of information transfer rate in them *(an analogue of the final speed of light according to the modern physical point of view)*. The given assumption plays a rather significant role in studies of the dynamic properties of ***d–CAs*** $(d \geq 1)$ and well meets the requirements of using models as the basis for modeling of parallel dynamic systems of various types and nature. This condition not only introduces a limitation on the speed of information propagation in ***CA***–models, but also determines the space *(a certain formal vacuum)* in which the dynamics of the development of the studied discrete objects, processes and phenomena occurs. At the same time, an arbitrary element of the alphabet ***A*** of the ***CA***–models can be selected as a resting state, but for a number of reasons we use element {***«0»,«□»***} as the most familiar and acceptable. ***CA*** models satisfying the above defining condition will be called *stable*, otherwise *unstable*. In the study of ***d–CA*** $(d \geq 1)$ models as independent mathematical objects, unstable models are also of certain interest. At the same time, unstable ***CAs*** may be of interest from the point of view of studying models in them, which are based on the concept of instantaneous transmission of information on arbitrary distances. Using of unstable ***CA***-models are not known to us, but the work in this direction seems quite interesting. We did not seriously consider this issue.

Thus, the ***dynamics*** of a classical ***d–CA*** $(d \geq 1)$ are fully defined in terms of ***LTF,*** i.e., local interactions of the neighborhood pattern automata of elementary ***z***–automata, whereas ***LTF*** $\sigma^{(n)}$ itself is a typical example of a local algorithm that functions in highly parallel manner, based on the states configuration of the local neighborhood elementary automata, that is determined by the neighborhood index ***X*** of the current ***z***–automata of

Z^d of the classical ***d–CA.*** The simultaneous application of the ***LTF*** to ***NP*** of each ***z***–automaton of the entire lattice Z^d defines the *global transition function* $\tau^{(n)}$ *(GTF)*, that converts the current ***CF*** $c \in C(A, d)$ of the lattice Z^d into a subsequent ***CF*** $c\tau^{(n)} \in C(A, d)$. Formally, the definition of the configuration $c\tau^{(n)}$ can be represented as follows. Let $C(A, d)$ be the set of configurations with respect to the lattice Z^d and alphabet ***A,*** and $s[z]$ denotes the current state of elementary ***z***–automaton; then, formally ***GTF*** $\tau^{(n)}$ with a neighborhood index $X = \{x_1, x_2, \ldots, x_n\}$ is determined by the following relationship, namely:

$$c\tau^{(n)} = c^* \leftrightarrow (\forall z \in Z^d)(s^*[z] = \sigma^{(n)}(s[z + x_1], s[z + x_2], \ldots, s[z + x_n]))$$

From this definition, it directly follows that each ***LTF*** $\sigma^{(n)}$ determines a single ***GTF*** $\tau^{(n)}$, and ***GTF*** $\tau^{(n)}$ can`t be determined by two different ***LTF*** $\sigma^{(n)}$. In other words, there is a one-to-one correspondence between the set of all ***GTF*** $\tau^{(n)}$ and the set of all ***LTF*** $\sigma^{(n)}$ for the given states alphabet ***A,*** dimension ***d*** of the lattice Z^d, and neighborhood index ***X.*** So, we can talk about ***GTF*** $\tau^{(n)}$, determined by ***LTF*** $\sigma^{(n)}$, and vice versa. It is proved that an arbitrary ***GTF*** in classical ***CA***–model is primitively recursive function [141]. This result determines not only the place of ***GTF*** τ in hierarchy of all recursive functions, but together with other components determines simplicity of mathematical objects, like cellular automata ***d–CA*** $(d \geq 1)$. Meanwhile, such simple ***CA***–models allow a rather complex dynamics of finite and infinite configurations, including universal computability.

We can now define the *fourth* component of ***d–CAs*** $(d \geq 1)$. For ***A,*** Z^d and ***X,*** the set of valid transforms ***T*** is any non–empty subset of the complete set of all ***GTF*** $\tau^{(n)}$ that are determined by three parameters ***A,*** Z^d and ***X.*** Besides, if set ***T*** contains one global transition function $\tau^{(n)}$, then object $d\text{–}CA \equiv \langle Z^d, A, \tau^{(n)}, X\rangle$ is called *monogenic* or *classical* ***d–CA*** $(d \geq 1)$. At that, the operation of a classic ***d–CA*** $(d \geq 1)$ is especially simple: if $c = c_o$ is an initial configuration of the homogeneous space Z^d at time $t = 0$ then configuration of the space Z^d at time $t = m$ is $c^* = c_o\tau^{(n)m}$ – the result of ***m***–fold application of the global transition function $\tau^{(n)}$ to configuration c_o of the homogeneous space Z^d of the ***CA***–model.

Let $\xi = \langle c_o\rangle[\tau^{(n)}]$ designates a sequence of configurations generated by some ***GTF*** $\tau^{(n)}$ from initial ***CF*** c_o. Then for finite ***CF*** $c_o \in C(A, d, \varphi)$ the sequence ξ represents the c_o configuration history in some classical ***d-CA*** $(d \geq 1)$ playing the main role in researches of dynamic properties of the classical ***CA***–models. Dynamics refers to the operation of a ***d–CA*** $(d \geq 1)$ of any type, which consists in changing over time the ***CF*** of the ***d–CA*** as a function of its initial configuration and the ***LTF (GTF)***. So, dynamics of ***d–CA*** $\equiv$ ***<Z^d, A, τ^(n), X>*** {*the sequence of configurations* $<c_o>[\tau^{(n)}]$; *the history of development of objects immersed in the* ***CA****–model*} is defined uniquely by the above basic five components ***d, Z^d, A, X*** and $\tau^{(n)}$ $\{\sigma^{(n)}\}$. Configuration $c^{-1} \in C(A,d)$ is the *immediate* predecessor for ***CF*** $c \in C(A,d)$ if $c^{-1}\tau^{(n)} = c$. Some configuration $c \in C(A, d)$ may have several immediate predecessors, their infinite number, or have no predecessors at all. At the same time, the immediate predecessors for *block*, *finite* and *infinite* ***CF*** in classical ***d–CA*** $(d \geq 1)$ are quite naturally defined in the obvious way [3]. In general, informally a *block* configuration refers to the configuration of the finite block of lattice Z^d, the *infinite* configuration contains an infinite number of elementary automata in states other than the rest state, finally the *finite* configuration is defined as a completely zero configuration of lattice Z^d with a block configuration immersed in it.

The task of actually determination predecessors for ***CA***-models is already laborious already for the case ***1–CA****s*. Below, we will deal mainly with *block* and *finite* configurations, since the case of infinite ***CF*** falls out of the scope of our attention, because it is possible to actually consider this type of ***CF,*** mainly, if they have a certain clear foreseeable structure, otherwise, as a rule, they have unpredictable dynamics; even the initial ***CF*** must be representable, for example, periodic in one aspect or another. A lot of software tools have been created for computer analysis of the existence of predecessors for block configurations [42]. In particular, we programmed tools for this purpose in *Mathematica* and *Maple* systems [60,62,64,65]. So, for the classical ***1–CA*** models, this problem is solved using *Mathematica* procedure [42], the call of which *Predecessors*[*c, f, n*] returns the list of predecessors for a *block* configuration ***c*** relative to ***LTF f,*** given by the list of parallel substitutions, to the depth ***n;*** if block ***CF c*** has no predecessors, the procedure call returns the empty list with output of the corresponding message. At the same time, this procedure turned out to be a fairly convenient tool in computer research of reproducibility of finite block configurations in classical one-dimensional ***CA***–models. It is easy to make sure that despite the simplicity of mathematical objects such as classical ***CA***–models, their dynamics are quite complex, and its research involves, in general, significant efforts, and in a number of cases includes the use of non–traditional approaches. For this reason, there are relatively few results obtained by theoretical methods in this direction, while a quite significant part of them were obtained through an empirical approach, including computer modeling [3,113,117,141-144,156-161].

Thus, the concept of classical ***d–CA****s* intuitively seems to us quite simple, in connection with which the question arises regarding its degree of the generality, that is, how widely such a concept allows extensions that do not exceed the scope of any studied phenomenon or the limits of some equivalence criterion *(some kind of stability property of the concept)*. A rather detailed analysis of a number of extensions of the classical ***d–CAs*** concept regarding its dynamic properties showed that, despite a rather strict equivalence criteria for the dynamics of two ***d–CA*** models *(based on a comparative analysis)*, the classical concept of ***d–CAs*** ***(d ≥ 1)*** has a sufficient degree of commonality, which allows us to consider it as one of the basis, determining the ***CA***-concept in its entirety. Having evaluated only generating capabilities of classical ***d–CAs (d ≥ 1)***, we have proved that a number of extensions of the classical concept of ***CA*** show that the concept of classical ***d–CAs*** has a quite sufficient degree of commonality with respect to the relatively narrow concept of equivalence of two ***d–CA*** models [141-144]. It follows from the definition of classical ***d–CA (d ≥ 1)*** that these objects represent formal parallel algorithms for processing of finite configurations from the set ***C(A, d, ϕ)*** by means of ***GTF,*** which can be considered as functions everywhere defined on the set ***C(A, d, ϕ).*** Of the aforesaid follows that the concept of classical ***d–CA*** $\equiv$ ***<Z^d, A, $\tau^{(n)}$, X>*** possesses a quite acceptable degree of community for many important applications *(despite all the simplicity);* is of considerable interest as an independent mathematical object, which is a very important component of a number of theoretical and applied models of parallel processing of information and calculations, including modelling of various natures.

Thus, if the three components ***Z^d, A*** and ***X*** of the cellular automata ***d–CAs (d ≥ 1)*** are sufficiently simple and transparent, then ***GTF $\tau^{(n)}$*** is, as noted above, a *primitive–recursive* function. Therefore, simple objects such as classical ***d–CAs*** have a sufficient degree of commonality and sufficiently complex dynamics to simulate a sufficiently extensive class of objects, processes and phenomena that occur in many fields. At the same time, these objects are of undeniable interest in study as an independent formal model of parallel processing and calculations. Meanwhile, within the framework of classical ***d-CA (d≥1),*** their special subclasses with specific characteristics such as ***CAs*** with refractory, memory and some others are chosen, which allow to more efficiently model a lot of fairly interesting objects and processes. We defined certain types of ***CAs***, which, however, were studied by us significantly less actively than classical ***CA***–models. For today, a number of extensions and generalizations of classical ***CAs*** models are used with varying degrees of intensity. However, not every extension of the classical concept of ***d–CAs*** models leaves us within the chosen equivalence criteria. According to the mentioned extensions of the ***CA***–model, a number of quite interesting results were obtained. Some of such ***CA*** types are discussed in detail in [3], other interesting types can be found in bibliographies [10,11,113,141-144,156-161,175,182-196].

The main object of our study within the framework of ***CA***–problematics were mainly classical cellular automata and their behavioural–equivalent modifications, for example, automata with refractory *(hereinafter we will call this group simply "cellular automata – **CA**").* Meanwhile, in general, the ***CA***–concept includes many types of automata *(polygenic, stochastic, non–deterministic, and others)* and in a number of cases we have studied or used some of these types in modelling. This applied primarily to the applied aspects of ***CAs*** problems, for example, in mathematical biology.

Meanwhile, the complexity and diversity of the real world do not fit at all into the Procrustean bed of the concept of classical ***CA***–models without any serious restrictions that affect the very attractiveness of its simplicity in its original concept. In our opinion, for today cellular automata are of undeniable interest in two main natural–scientific fields, namely:

*(**1**) The modeling environment and the embending in it of a wide variety of processes, objects and phenomena (especially those that are difficult or impossible to describe by other means, in particular, based on partial differential equations); that is, today the largest number of researches has been made in this direction and rather interesting results have been obtained for today;*

*(**2**) As an independent mathematical object of research (highly parallel dynamic discrete systems; highly parallel computers similar to the Post, Markov systems and Turing machines, etc. for sequential calculations; text processing systems with highly parallel substitution rules, etc.).*

In this section, we have informally introduced the concept of classical *cellular automata (**CAs**)* and related concepts, thereby defining the main object of our research. Purpose of defining this object is to introduce in ***CA***-problematics of the reader, above all, who is previously little familiar or completely unfamiliar with it. A detailed analysis of the definitions and concepts entered is not given here, however the interested reader can familiarize it, for example, in [156-161]. Naturally, the definitions and concepts introduced do not cover all ***CA***-problematics and as the material is presented, other necessary concepts, definitions and designations will be introduced. Consideration of the presented material is conducted at an informal, substantive level, while their rigorous consideration is available from the sources cited. This approach is due to the fact that our goal is to informally represent our activity in ***CAs*** issues, and not in strict evidence of results obtained by us. This setting method is more like a surwey that, in our opinion, allows to delve more into the proposed problem without complicating it with evidence, at times, quite voluminous and requiring additional information from certain other subject fields.

5.2. The nonconstructability problem in classical cellular automata

By the terms *nonconstruction configuration (NCF)* **&** *nonconstructability* we, as a rule, mean block configurations such as *"Eden Garden"* and the presence in ***d–CA (d≥1)*** of such configurations, respectively; the concept of *nonconstructability* defines one of the fundamental characteristics of ***d–CA,*** consisting in the presence of configurations for them that can`t be generated at the moment $t > 0$ from an arbitrary ***CF*** at the initial moment $t = 0$. Meanwhile, the *nonconstructability* problem has a slightly broader understanding, which can generally be characterized as follows.

First of all, regarding the classical ***d–CAs (d≥1)***, we are dealing with two sets of essentially different configurations: *finite* configurations ***C(A,d,ϕ)*** and *infinite* ***C(A,d,∞);*** collectively, these sets constitute the set ***C(A, d)*** of all configurations. The conventional concept of *nonconstructability* is directly related to the impossibility of generating from any configuration $c \in C(A,d)$ by ***GTF*** of a classic ***d–CA (d≥1)*** of a configuration containing a certain *block* configuration. The fundamental difference between *finite* and *infinite* configurations in the case of classical ***d–CAs (d≥1)*** allows to quite naturally differentiate this concept of *nonconstructability*, which provides a more detailed study of the dynamics of classical ***CA***–models along with a number of results that are rather fundamental. In particular, along with nonconstructible block configurations, it is quite appropriate to study the nonconstructability of finite configurations related to both the set ***C(A, d)*** in general and the subset ***C(A, d, ϕ).*** This approach allows natural introduction of two new concepts of nonconstructability, namely ***NCF–1*** and ***NCF–2,*** that are not equivalent to both each other and with the standard ***NCF*** concept. Along with the generally accepted concept of nonconstructability, we have identified and considered some other rather important concepts of nonconstructability, including the above ones [4,5].

In general, ***reversibility*** is a rather multiaspect concept. For the classical ***d–CA (d≥1),*** that are a subclass of parallel discrete dynamic systems, the question of studying the *reversibility* of dynamics *(trajectories)* of finite configurations seems interesting and quite natural. It is natural to assume that a configuration $c \in C(A, d, \phi)$ has the *reversible* dynamics if for it c^p configuration *(direct or indirect predecessor)* is the only one, where $p \in$ {***-1, -2, -3, ...***} and $c^p \tau^{(n)} = c^{p+1}$, $c^p \equiv c$. However, under this condition, we have two alternatives: *(1) a* c^p *configuration should belong only to the set* ***C(A, d, ϕ)*** *or (2) the set* ***C(A, d, ∞).*** With this in mind, we have defined a number of concepts of *reversibility* that allows us to more fully consider this concept regarding classical ***CAs***. We define the concepts of ***real*** and ***formal*** *reversibility* because of the two main types of nonconstructability for classical ***CA***–models *(types **NCF** and **NCF–1**)*.

The *nonconstructability* issues of configurations are fundamental in the mathematical theory of ***CA***–models along with their many applications, especially when using their applied and conceptual models of spatially distributed discrete dynamical systems, of which real physical systems are the most preferred prototypes. It is for this reason that this problem raises questions of considering the theoretical aspects of classical ***CA***–models. The nonconstructability problem is of a serious gnoseological interest in the case of embedding in ***CA***-models of different cosmological objects. This may be due to various aspects of the reachability problem of certain conditions or clusters in the formation of special cosmological objects. At the same time, the reversibility of the main physical processes can serve as an analogue of the absence of certain nonconstructability types in classical ***CA***–models [141]. This problematics is becoming more and more relevant in terms of formation of modern physical theories, and in connection with a number of attempts to interpret different anomalous phenomena from traditional points of view.

Nonconstructible configurations types*.** The definition of the basic type of *nonconstructible configurations (**NCF**)* goes back to *E.F. Moore* and *J. Myhill* [1,3] concerning the *block* configurations, i.e. a configuration $c_b \in C(A, d, W)$ of a finite d–dimensional W–hypercube of automata in Z^d $(d \geq 1)$ is an *nonconstructable configuration (**NCF**)* if and only if there is no configuration $c \in C(A, d)$ such that $c_b \subset c\tau^{(n)}$. The nonconstructability ***NCF with respect to set of all finite configurations is equivalent to the existence of such configurations c from $C(A, d, \phi)$ for which there are no predecessors from the set $C(A, d)$. This nonconstructability concept is the strongest *(it can be called "absolute" to a certain extent)*. However, at one time this moment caused a lot of discussions and misunderstandings, so such concept of nonconstructability in classical ***CA***–models was by us analyzed in detail and differentiated from the point of view of essence of classical ***CA***–models [141-144,156-161]. In general, the ***NCF*** definition represents a generalized concept of nonconstructability at level of block configurations and finite ones, quite naturally identifying both concepts. At the same time, the second approach to the nonconstructability concept of ***NCF*** seems to us more preferable from the point of view of studying of different aspects of the classical ***CA***–models dynamics.

In view of differentiation of the set $C(A, d)$ on 2 not intersecting subsets $C(A, d, \phi)$ and $C(A, d, \infty)$ is quite natural for us to differentiate the general nonconstructability problem of finite configurations for case of classical ***CA***–models relative to these subsets that is quite visually illustrated by the following table that is rather transparent and doesn't demand special explanations, at the same time, clarifying the basic essence of the issue.

Availability of predecessors for a configuration $c \in C(A, d, \phi)$		
$C(A, d, \infty)$	$C(A, d, \phi)$	***Type of nonconstructability***
–	–	***NCF***
+	–	***NCF–1***
–	+	***NCF–2***
+	+	***Absolute constructability***

The table at the informal level defines and depletes the basic types of nonconstructability of finite configurations in classical ***CA***–models while the issue of nonconstructability of infinite configurations went beyond our consideration, primarily due to their poorly developed principles of processing, interpretation and formation. At the same time, the infinite configurations, such as one–dimensional configurations, may be studied in connection with their ability to quite satisfactory represent numerical or other well–interpreted objects [141-144]. Based on the sets of finite, block and infinite configurations, we were able to significantly advance both the differentiation and detailing of the nonconstructability concept in classical ***d–CA****s* ***(d≥1)*** relative to the previous state of this question. It is enough to make sure that the nonconstructability concept such as ***NCF*** refers primarily to block configurations, which allows us to consider the nonconstructability of two classes, usually not equivalent to each other: *(**1**) block nonconstructability* and *(**2**) configuration nonconstructability in classical **CA**–models*. It is easy to make sure: *If a classical model **d–CA (d≥1)** possesses the nonconstructability of the **NCF** or/and **NCF–1** type, then the **NCF** and **NCF–1** will be infinite disjoint sets of corresponding configurations; if a classical **CA**–model does not possess **NCF,** then it will possess **NCF–1** and/or **NCF–2.*** The nonconstructability problem for classical models ***d–CA (d≥1)*** was investigated in sufficient detail by us and a number of interesting results in this direction were obtained [141-144,156-161]; some of them will be presented below. The results were both qualitative and quantitative. At the same time, the lion's share of the results belonged to ***1–CA*** models, whose studies are significantly simpler than ***d–CA*** models ***(d > 1)***, primarily due to the dimension of the models, which plays a decisive role in the study of many essential properties of the dynamics of classical ***CA***–models. In particular, on the basis of a numerical approach, we have proved [3] that the problem of determining of the nonconstructability for classical models ***1–CA****s* is algorithmically solvable, while for models ***d–CA****s* ***(d > 1)*** this problem is algorithmically unsolvable. As our studies of the dynamics of classical ***CA***–models [141-144] showed, the dimensionality significantly differentiates many aspects of the classical ***CA***–models dynamics, often complicating their.

Thus, *block* nonconstructability of the ***NCF*** type causes *configuration* nonconstructability, while the opposite, generally speaking, is incorrect. In this regard, we have identified a new type of nonconstructability that occurs at the boundary of block and configuration nonconstructability, allowing for its qualitative expansion. In this case, the nonconstructible configurations of the ***NCF–3*** type are defined as follows:

A configuration $c^* = \square c_b \square \in C(A, d, \phi)$ *is nonconstructible of* ***NCF–3*** *type if and only if block configuration* c_b *of* ***d****–dimensional hypercube of unit automata in* ***d–CA*** *(****d ≥ 1****) is constructible, however configuration* ***c**** *is nonconstructible, where* □ *is the environment of the block configuration* c_b *by infinite number of resting states "****0****".*

Note that the presence of nonconstructible configurations of type ***NCF–3*** in the ***CA***–models to a certain extent determines a somewhat unexpected result: *If there is a constructible kernel (non-zero part) in some finite* ***CF,*** *the configuration itself can be nonconstructible one*. So, the presence of nonconstructability ***NCF–3*** for a ***CA***–model necessarily entails presence and ***NCF,*** while the opposite, generally speaking, is incorrect. The above four types of nonconstructible configurations *(****NCF, NCF–1, NCF–2*** *and* ***NCF–3****)* are pairwise non-equivalent and allow more detail to investigate the nonconstructability problem in classical ***d–CA****s (****d ≥ 1****)* models. The nonconstructability of type ***NCF–3*** can be considered as some subclass of the general nonconstructability ***NCF*** which in some cases is a quite certain interest in theoretical and applied studies of classical ***CA***–models. First of all, this concerns the study of models as formal parallel systems for processing of finite words in finite alphabets, as well as modeling at the formal level some processes, including computational processes. A detailed discussion of this concept of nonconstructability is presented, for example, in [113,141-144,161,182-196] and in many others works.

Meanwhile, it should be noted that nonconstructability of type ***NCF–3*** is rather narrow and was defined by us as a result of the study of dynamics of infinite configurations having special structures [141]. This type of nonconstructability does not belong to the main types mentioned above and we, in practice, did not pay much attention to it. By identifying ***NCF, NCF–1,*** and ***NCF–2*** as basic types of nonconstructability in classical ***CA*** models, we identified and nonconstructability types such as ***NCF–3*** and relative nonconstructability *(relative to a certain subset of set* ***C(A, d, ϕ)****)*. This allowed us not only to study substantially more in detail the essence of nonconstructability in ***CA***–models, but also to obtain quite strong tools for studying many dynamic properties of classical ***CA***–models.

If a certain ***NCF*** *(****NCF–3****)* is *absolutely* nonconstructible configuration relative to the set ***C(A, d, ϕ)*** ∪ ***C(A, d, ∞),*** then the configurations ***NCF–1*** and ***NCF–2*** are relatively nonconstructible configurations relative to the sets ***C(A, d, ϕ)*** and ***C(A, d, ∞)*** respectively. In the following table, the sign "+ (–)" indicates the presence *(absence)* of the corresponding type of the nonconstructible configurations in the classic ***d–CA (d ≥ 1)***, determining the permissible combinations of their types. Detailed discussion of these and related questions from different points of view *(both quantitative and qualitative)* can be found, for example, in [113,141,161,182-196].

Valid nonconstructability types for classical models d–CAs (d ≥ 1)				***Admissible types combinations***
NCF	***NCF–1***	***NCF–2***	***NCF–3***	
+	+	+	+	***Yes***
+	+	+	–	***Yes***
+	+	–	+	***Yes***
+	+	–	–	***Yes***
+	–	+	+	***Yes***
+	–	+	–	***Yes***
+	–	–	+	***Yes***
+	–	–	–	***Yes***
–	+	–	+	***No***
–	+	–	–	***No***
–	–	+	+	***No***
–	+	–	–	***Yes***
–	–	+	–	***Yes***
–	–	+	–	***Yes***
–	+	+	+	***No***
–	+	+	–	***No***
–	–	–	+	***No***
–	–	–	–	***No***

So, it follows from the table that the classical models ***d–CA (d ≥ 1)*** have at least one type of nonconstructible configurations ***NCF, NCF-1, NCF-2*** and/or ***NCF–3.*** The nonconstructability problem as a whole, including a number of special cases, was investigated in detail by us regarding the classical models ***d–CA (d ≥ 1)*** at both the formal and experimental *(based on computer modeling in Maple and Mathematica systems)* levels. The results obtained in this direction were used by us to research the dynamic properties of ***CA***–models, which revealed a number of quite interesting behavioral properties of these models. These results are available at both formal and qualitative levels in [113,141-144,156-161,182-196].

The nonconstructability criteria for classical CA–models. As the first criterion for existence of ***NCF*** in classical ***CA***–models, the *Moore–Myhill* criterion can be considered based on *mutually erasable configurations* ***(MEC)*** defined as follows. Let ***W*** be a coherent block of unit automata of a ***d–CA*** model ***(d ≥ 1)***, and ***B*** be a set of all neighboring automata for ***W*** according to a neighborhood index ***X.*** Let ***CF(P)*** now be a configuration of a finite block ***P*** of elementary automata of the model. Then the block configurations $CF(B) \cup CF(W_1)$ and $CF(B) \cup CF(W_2)$ are referred to as the ***MEC*** pair for a global transition function $\tau^{(n)}$ in ***d–CA*** if and only if:

$$[CF(B) \cup CF(W_1)]\tau^{(n)} \equiv [CF(B) \cup CF(W_2)]\tau^{(n)}; \qquad CF(W_1) \neq CF(W_2)$$

Block ***W*** will be referred to below as an internal block ***(IB)*** of the ***MEC*** pair and denote ***IBMEC.*** While in the case of finite configurations c_1, c_2 $(c_1, c_2 \in C(A, d, \phi))$ they form the ***MEC*** pair if and only if $c_1 \neq c_2$. Definition of ***MEC*** pairs for finite configurations is quite convenient in the research of the classical ***CA***–models dynamics. Obviously, both ***MEC*** definitions are equivalent relative to *erasability* property for classical ***CA***–models.

At one time, quite a few interesting questions were formulated regarding ***MEC*** by *E. Moore*, the solution of which made it possible to obtain a lot of interesting results for the ***1***–dimensional case. In particular, our results illustrated all variety of ***IBMEC*** types even in the case of rather simple binary classical models ***1–CAs*** with the *Neumann–Moore* neighborhood index [3]. We shown that many results related to the nonconstructability problem form a fairly effective part of the basic tools for studying the dynamics of classical ***CA***–models, so various estimates of ***IBMEC,*** along with other aspects of this problematics, are of undeniable interest. In this regard, relatively to important question of the minimum size of a simple ***IBMEC*** going back to *E. Moore*, we obtained the following result [3]:

For integers $a \geq 3$ and $n \geq 2$ in the classical 1–CAs with states alphabet $A = \{0,1, ..., a–1\}$ and neighborhood index $X = \{0,1,2, ..., n–1\}$ there are MEC with a prime IB of minimum size n. Fraction $\Delta(a, n)$ of classical 1–CA with IBMEC of minimum size one, with respect to all 1–CAs with alphabet $A = \{0,1, ..., a–1\}$ and neighborhood index $X = \{0,1, ..., n–1\}$, satisfies the relation $\Delta(a, n) > (2a^n – 1)/a^{2n}$. If in a classical 1–CA with alphabet $A = \{0,1, ..., a–1\}$ and neighborhood index $X = \{0,1,2, ..., n–1\}$ there are IBMEC of minimum size f, then the ratio $1 \leq f < a^{n-1}(a^{n-1}–1) + n–2$ takes place. The problem of determining the minimum size MEC for d–CAs ($d \geq 2$) is algorithmically unsolvable.

We considered the issues related to the existence of ***MEC*** in detail and in this direction we obtained a number of rather interesting results [3,5,113, 117,141-144,156-161]. These studies were driven by the fact that the first and one of the main criteria for the existence of ***NCF*** in ***CA***–models *(and not only classical ones)* is the *Moore–Myhill* criterion:

An arbitrary classical model d–CA (d ≥ 1) has NCF if and only if there are MEC pairs for its global transition function.

In the future, this criterion was somewhat generalized by us taking into account cases of nonconstructability of types ***NCF–1*** and ***NCF–3*** [144]. Our research in this direction has yielded a number of both *quantitative* and *qualitative* results, revealing many issues of nonconstructability in classical ***CA***–models [3,5,10,11,113,117,141-144,156-161,182-196].

Meanwhile, rather detailed studies of the nonconstructability problem in classical ***CA****s* led us not only to understand the insufficient effectiveness of approach based on the ***MEC*** concept, but also allowed us to introduce the ***γ***–configurations concept ***(γ–CF)*** [165], which turned out to be quite fruitful. Our concept of ***γ–CF*** is slightly different from the concept of ***k***–balanced global transition functions introduced independently of us by *A. Maruoka* and *M. Kimura* [166] in *1976* but is completely equivalent to it. The ***γ***–configurations concept was introduced by us as part of research of the nonconstructability problem in classical ***CA***–models while by *Kimura* and *Maruoka* for study of parallel global mappings ***τ: C(A, d) → C(A, d),*** determined by global transition functions ***τ*** in the classical ***d–CA****s*. Let us introduce the very concept of block ***γ***–configurations ***(γ–CF****s)*.

Let ***#(b)*** *be the number of elements of an arbitrary set* ***b*** *and* ***CF(j)*** *be the set of all kinds of configurations of the finite block* ***j*** *in the alphabet* ***A*** *of the classical cellular model* ***CA(A, d, τ, X).*** *Let's say that in a classical* ***CA***–*model* ***CA(A, d, τ, X),*** *the* ***γ–CF****s exist on a finite* ***g***–*block if and only if* ***g(γ) ≠ #(CF(g ∪ B))/#(CF(g))*** *of predecessor–configurations (relative to* ***GTF*** *of the* ***CA***–*model) exist for at least one configuration of* ***g,*** *where* ***B*** *is block of elementary automata which are adjacent to all automata of the* ***g***–*block according to the neighbourhood index* ***X*** *of the* ***CA***–*model.*

Based on the ***γ–CF*** concept, we obtained a new criterion for the existence of ***NCF*** in classical ***CA***–models [165], of significant interest for research on ***CA****s* issues in general, especially their dynamic aspects.

A classical d–CA model (d ≥ 1) has nonconstructability of the NCF type (possibly NCF–3) if and only if γ–CFs exist for it. Any CA–model will possess the MEC if and only if the model possesses γ–configurations.

Note that this statement refers to both classical and unstable ***CA***–models. The ***γ–CF*** concept turned out to be significantly more effective than ***MEC*** in obtaining of various numerical estimates for the classical ***CA***–models, allowing often to obtain estimates quite close to the optimal [141-144].

Indeed, the criterion based on *γ*-configurations allowed us to obtain much more acceptable estimates for certain numerical characteristics of ***d–CAs (d ≥ 1)***. In particular, quite significant contrast can be obtained in results of this type regarding the use of the concepts ***MEC*** and ***γ***–configurations. So, an example of the application of both approaches for estimating the minimum dimensions of ***NCF*** in relation to the known game *"Life"* [161] is very clear. It's easy to make sure that the game is nothing more than a binary classic ***2-CA*** with *Moore* neighborhood index. Study on the *"Life"* game was conducted by many mathematicians and programmers along with many amateurs, mainly based on computer modeling. This classical ***2–CA*** model is one of the most famous. *A.R. Smith*, studying the binary ***2–CA*** corresponding to the *"Life"* game, showed that this ***CA***–model has minimum size *1010x1010* of ***NCF.*** To obtain this an estimate, *A.R. Smith* used an approach based on the ***NCF*** concept, while based on the concept of ***γ***–configurations, we managed to significantly improve this estimate, lowering it to a quite visible size *49x49*. In this direction there are many other results of this type [141-144]. So, the ***γ***–configurations concept of *Aladjev–Kimura–Maruoka,* along with the nonconstructability criterion based on it, allows to enough effectively investigate a lot of quantitative aspects of dynamics of classical ***d–CA*** models, while the ***MEC*** concept in a number of cases is more convenient for their qualitative study. Thus, in many ways, both concepts complement each other quite well.

The same way we investigated universal classical binary model ***2–CA*** of *E. Banks* [164] which was at the time minimum on complexity. It can be shown that this ***2–CA*** model is suitable for implementing computational schemes of arbitrary complexity in it. We have proved [141] that there are ***NCF*** in the model already on blocks of size *14x14.* In this regard, a rather interesting hypothesis arose:

The universal classical models d–CA (d ≥ 1) with minimal complexity d*(states alphabet cardinality)*(neighborhood template size) will have the nonconstructability of NCF and/or NCF–1 type.

Today, all the minimum universal ***d–CAs (d ≥ 1)*** known to us correspond to this hypothesis. An extensive enough bibliography on this issue can be found in [11,117,141-144,156-161,164]. The same question includes our numerical results concerning on the sizes of ***MEC, γ–CF, NCF, NCF–1, NCF–2*** and ***NCF–3*** along with some other important characteristics.

In order to better understand the nonconstructability concept and to create on its basis an effective apparatus for study the dynamics of ***CA***–models, it is extremely desirable to identify certain relationships between various characteristics of ***MEC, γ–CF, NCF, NCF–1, NCF–2*** and ***NCF–3*** both quantitative and qualitative ones. We considered this issue in sufficient detail, which made it possible to get many of interesting results [141-144, 156-161]. So, especially many results *(both quantitative and qualitative)* regarding ***MEC, IBMEC, γ–CF, NCF, NCF–1, NCF–2*** and ***NCF–3*** were obtained by us for classical ***1–CAs*** models; a number of those results had proved to be exhaustive decisions on one question or another. Among them, one can note in particular the result that determines the estimate of the maximum size of the internal block of ***MEC*** for an arbitrary ***1–CA*** model, which is the basis for proving the algebraical solvability of the problem of the existence of ***MEC, γ–***configurations, and therefore ***NCF (NCF–3)*** for such models. While on the basis of the unsolvability of the *domino* problem, we proved the algebraic unsolvability of the existence problem of ***MEC, γ–CF,*** and ***NCF (NCF–3)*** for ***d–CAs*** $(d \geq 2)$ models. Further research on this issue allowed us to present a new ***MEC*** concept as a definite basis for the generalized nonconstructability criterion in the classical ***d–CAs*** $(d \geq 1)$ models [4,5,10,11,113,141,183], namely:

Two configurations $c_1, c_2 \in C(A, d)$ $(c_1 \neq c_2)$ ***form the pair of generalized mutually erasable configurations (MEC–1) with respect to the global transition function*** $\tau^{(n)}$ ***of a classical model d–CA*** $(d \geq 1)$ ***if and only if the next determining relationship*** $c_1\tau^{(n)} = c_2\tau^{(n)} = c^* \in C(A, d, \phi)$ ***is valid for them.***

Pairs of ***MEC–1*** similar to ***MEC*** pairs may be formed by configurations such as ***NCF–1*** and/or ***NCF,*** i.e. {***NCF–1, NCF–1***}, {***NCF, NCF***} and {***NCF, NCF–1***}. Unlike ***IBMEC,*** a certain analogue is defined for the ***MEC–1*** in the form of a *"absorption node"* ***c*,*** whose size is of certain interest in a number of numerical studies of nonconstructability, which is due to the ***MEC–1*** pairs existence. The expediency of this concept was due to the fact that the of ***MEC–1*** pairs definition allows to use infinite configurations too. For example, it has been shown that the minimum sizes of configurations ***NCF–1, NCF*** and ***c**** can be identical. At that, if ***c**** can be ***NCF–1,*** then ***c**** can`t be ***NCF;*** property is based on principal difference between the nonconstructability types ***NCF*** and ***NCF–1.***

Meanwhile, there are a number of other differences between the concepts ***MEC*** and ***MEC–1.*** So, the presence of ***MEC*** for classical ***d–CAs*** $(d \geq 1)$ model entails the presence of ***MEC–1*** in the model, while the opposite is generally incorrect. This circumstance is caused by the fact that the ***MEC*** presence in arbitrary classical model ***d–CA*** $(d \geq 1)$ is one of two criteria of existence of the ***NCF*** nonconstructability in the model, whereas the existence in the model of ***MEC–1*** not necessarily cause the presence of ***NCF*** nonconstructability for ***CA–***model. The above concept of mutual erasability ***MEC–1*** in the classic ***d–CAs*** $(d \geq 1)$ models is closely related to the general nonconstructability problem as evidenced by the following important enough result:

A classical d–CA ($d \geq 1$) model has at least nonconstructability of the NCF type (possibly NCF–3) or NCF–1 if and only if MEC–1 pairs exist for the model. If MEC–1 there are no for the model d–CA ($d \geq 1$), the model will have NCF–2; moreover, the existence of NCF–2 can be fully combined with the existence for the model of MEC–1.

This result is a significant enough generalization of the *Moore–Myhill* criterion *(based on **MEC**)* and the *Aladjev–Kimura–Maruoka* criterion *(based on γ–configurations)* that is equivalent to the first, extending them to other types of nonconstructability in ***d–CAs*** ***($d \geq 1$)*** models. A number of important results on ***CAs*** dynamics were derived from this criterion. Our results on the nonconstructability questions for classic ***d–CA*** models ***($d \geq 1$)*** in general can be found, in particular, in publications [3-5,10,11, 113,141-144,155-161,164,182-196] and in references cited in them.

Note, along with classical ***CA***–models, the so–called *finite* ***CA***–models, which consist of any, but finite number of elementary automata, are of essential application interest. This class of ***CA***–models from a theoretical standpoint was studied quite intensively by the *Japanese* school, as well as by a number of other researchers [11,164]. Whereas our results in this direction are quite limited and are presented in [11,141-144]. Meanwhile, study in this direction is quite promising, taking into account numerous applied aspects of this class of ***CA***–models and, first of all, using them as parallel discrete models of various processes, phenomena and objects. In particular, it is shown that in the general case of finite ***CA***–models, the presence of ***MEC*** pairs may be sufficient, but not necessary, for existence in them of nonconstructability of the ***NCF*** type. Moreover, the problem of nonconstructability for finite models is very closely related to the type of boundary conditions [161]. Many of very interesting properties of the finite ***CA***–models were obtained on the basis of computer simulating in *Maple* and *Mathematica* systems [10,11,50-60,66-73,113,182-196].

Another interesting class is the so–called *cellular automata on partition* ***(CAoP)***, introduced by *T. Toffoli* and *N. Margolus*. The ***CA–model on the partition*** is defined as an ordered tuple of five basic components ***CAoP*** ≡ $\langle Z^d, A, m, \Psi^{(h)}, \Xi \rangle$, where the first two components Z^d and ***A*** are similar to the case of classical ***CA***–models, ***m*** is the edge size of ***d***–dimensional hypercube into which the Z^d space is broken; $\Psi^{(h)}$ – local block transition function ***(LBF;*** $h=m^d$***)***; **Ξ** – rules of switching of blocks of the Z^d space. The functioning of models ***d–CAoP*** ***($d \geq 1$)*** is sufficiently simple and is considered sufficiently detailed in [113,141]. Some comparative analysis of the models of both types *(**CAs** and **CAoPs**)* was also carried out there. Currently, models like ***CAoP*** are widely used, primarily to solve a lot of physical modeling problems, using, in particular, software and hardware of *CAM* machines based on ***CA****s* computing models [113,164,183].

The ***CAoP*** models and some their interesting modifications were studied by us both by theoretical and simulation methods. In the second case, a number of procedures for *Mathematica* were programmed, which made it possible to experimentally investigate a number of significant aspects of the ***CAoPs*** and a number of their modifications *(composition of models, dynamics of models, etc.)* [161]. First of all, by us has been shown that for models ***CAoP,*** the classification of nonconstructability, like the case of classical ***CA***–models ***(NCF, NCF–1, NCF–2, NCF–3)*** is not entirely appropriate. A criterion for the nonconstructability existence of ***NCF*** type for ***CAoP*** models was established, which can be formulated as follows:

A model CAoP will have NCF nonconstructability if and only if MEC pairs exist for such model in the above sense.

It is shown [141] that number of the models $\boldsymbol{CAoP \equiv \langle Z^d, A, m, \Psi^{(h)}, \Xi\rangle}$ which don't possess the ***MEC*** pairs and, therefore, ***NCF*** is equal to $\left(a^{m^d}\right)!$ while their share concerning all such models will be equal to $\left(a^{m^d}\right)! \big/ a^{m^d}a^{m^d}$, i.e. fast enough approaching zero already with sufficiently small values ***a, m*** and ***d.*** Thus, in the class of ***CAoP*** the models that to a certain extent have the *reversibility* property are *"exotic"*. At the same time the absence of ***NCF*** nonconstructability for some model ***CAoP,*** entails the closure of the set $\boldsymbol{C(A, d, \infty)}$ relative to the global transformation $\boldsymbol{\tau^{(h)}}$ of the model, therefore the absence of ***NCF–1*** for it. While for classical ***CA***–models, this statement is generally incorrect.

The closure problem of the set $\boldsymbol{C(A, d, \infty)\ (d \ge 1)}$ relative to the global $\boldsymbol{\tau^{(h)}}$ transformation, determined by local block function $\boldsymbol{\Psi^{(h)}}$ of the ***d–CAoP*** model, is algorithmically solvable whereas a set of ***NCF*** for any ***d–CAoP*** model is recursive. The closure of the set $\boldsymbol{C(A, d, \infty)}$ with respect to the mapping defined by local block function $\boldsymbol{\Psi^{(h)}}$ of a model ***d–CAoP*** $\boldsymbol{(d \ge 1)}$ results in the presence of the nonconstructability ***NCF*** for the model, but the inverse statement is generally speaking incorrect. Therefore, for the ***CAoP***–models the existence of ***NCF–1*** without ***NCF*** is impossible. On the other hand, for the classical ***CA***–model, the nonconstructability types ***NCF*** and ***NCF–1*** are not equivalent, in the absence of ***NCF*** for it, this model can have ***NCF–1.*** If problem of existence of nonconstructability ***NCF*** for classical ***d–CA****s* $\boldsymbol{(d \ge 2)}$ is algorithmically unsolvable whereas in the class of ***d–CAoP****s* $\boldsymbol{(d \ge 1)}$ the problem is algorithmically solvable and the algorithm of the constructive decision reduce to clarification of lack / existence of mutual unambiguity of mapping $\boldsymbol{\Psi^{(h)}: A^h \Rightarrow A^h}$. It is shown [141] that ***in general, the nonconstructability property of NCF type with***

respect to the mutual modeling of the d–CA and d–CAoP models is not invariant. Moreover, modeling of an irreversible model by an appropriate reversible model is quite acceptable [10,11,113,141,183].

It follows from the arguments [141] that based on the same definition of nonconstructability of the ***NCF*** type, we obtain that its cause-effect bases for classical ***CA***–models and ***d–CAoP***–models differ significantly. This *difference* underlies the major *differences* for many fundamental dynamic properties of classical ***CA***–models and ***CAoP***–models and gives rise to a significantly greater need to model processes that need the *reversibility* property of their dynamics [113,141,161]. By completing a brief survey of some our results *(details of our results in this direction can be found, for example, in* [148,155-161]) regarding the general nonconstructability problem in classical ***CA***–models, we will focus on its features due to the *reversibility* problem of models, which is quite important both theoretical and applied interest. At the same time, we understand the *reversibility,* taking into account the specifics of the classical ***CA***–models, explaining its specificity as follows.

The reversibility problem of classical CA–models. The ***reversibility*** of classical ***d–CAs*** models ***(d ≥ 1)*** is one of the most important properties, primarily from the standpoint of modeling of various physical processes and calculation theory; it is closely related to the nonconstructability of ***NCF*** type for the ***CA***–model, primarily. At the same time, some remarks of a principled nature regarding the problem of *reversibility* in general, which, in turn, is very closely related to the nonconstructability problem for ***CA***–models in general and for classical ***CA***–models in particular, are quite relevant here. On the formal level, the *reversibility* problem of the function ***F*** from ***n*** variables {***x1, x2, ..., xn***} is reduced to the question of the possibility of unambiguous recovery for it of any tuple {***x1, x2, ..., xn***} based on the known form of the function ***F*** and its value ***F(x1, x2, ..., xn)*** on this tuple. Naturally, on ***n*** inputs and ***(n–k)*** outputs of some algorithm, provided that they belong to the same alphabet, it is impossible to obtain this reversibility type {***k = 1..n–1***}. Therefore, along with the result ***F(x1, x2, ..., xn),*** it is required to have ***(n–1)*** values of tuple ***<x1, x2, x3,...,xn>*** to restore the missing value x_j; $j \in \{1, 2, ..., n\}$, that is, we should have some additional information that allows, based on the type of function ***F,*** along with its value on the tuple, to restore the entire desired tuple. In principle, some other methods of obtaining such additional information quite may be used. A number of approaches have been proposed to create reversible computer models, including at the biomolecular and chemical levels. In [141,113,161] a quite detailed discussion of the *reversibility* problem as a whole is given. Regarding ***CA***–models, the problem has some specificity.

In general, by ***reversibility*** in the case of classical ***CA***–models, we mean two types of it: ***(1)*** *block reversibility* and ***(2)*** *configuration reversibility*. In both cases, the predecessor of the corresponding type *(one, more or not)* must be on the previous step of the ***CA*** model progress. In the case of *block* configuration, *block* configurations are the predecessors, while in the case of *finite* configuration, *finite* configurations must be, in the opposite case the finite configuration is relied by us as a nonconstructible one of type ***NCF–1*** or/and ***NCF.*** Therefore, a classical ***CA***–model can be *block*–reversible and *configurationally* irreversible. Obviously, a finite configuration containing the nonconstructible block subconfiguration is irreversible. This approach to the reversibility concept plays an important role in the study of both theoretical and applied aspects of the dynamics of classical ***CA***–models. Meanwhile, the use of ***CA***–models as formal and promising prototypes of computing systems involves the research of the questions concerning the *reversibility* of the dynamics of these models.

The *reversibility* problems of ***CA****s* dynamics play a rather important role, primarily in terms of their using as an environment for modeling various physical processes. In this regard, one of the main research issues in the classical ***CA***–models is the reversibility of their dynamics. Today there are a number of quite interesting classes of ***CA***–models with the general reversibility property, among which the aforementioned ***CAoP***–models can be noted, together with reversible ***CA***–models specially developed by *T. Toffoli* and studied from standpoint of computational and constructive universality. Quite a lot of study is devoted to various questions of the ***CA***–models reversibility of of different types and classes [113,141,164]; [161] presents the most interesting results in this direction. Meanwhile, the reversibility problem for classical ***CA***–models is more multifaceted and is discussed in [113,161]. In general, the reversibility problem of the ***CA***–models is not so unambiguous. Meantime, we used a stricter notion of reversibility, which we understand as a possibility of unambiguously restoring the dynamics of ***CA*** at any time; that is, such reversibility when it is possible to accurately determine at each moment $t > 0$ for each finite configuration in the ***CA***–model of its sole predecessor at moment $t - 1$.

A d–CA ≡ <Z^d, A, $\tau^{(n)}$, X> dynamics we will call reversible in only case when for each configuration $c \in C(A, d)$ exists the only one predecessor $c^* \in C(A, d)$ such that $c^*\tau^{(n)} = c$; otherwise, the dynamics of such model will be called irreversible.

Thus, the dynamics irreversibility for a ***CA***–model is naturally defined by the absence of predecessors for some configuration of $c \in C(A, d)$ or the presence of more than one predecessor from the set $C(A, d)$ for it. At the same time, as noted above, for classical ***CA***–models, irreversibility can be both *block* and *configuration*. The presence in classical ***CA***–model of the ***NCF–1*** nonconstructability even in the absence of ***NCF*** leads *(according to our concept of reversibility)* to the irreversibility of such model, more precisely, the dynamics of its configurations in general [3,141,161]. Our definition of irreversibility can be considered as purely formal because of the uniqueness of infinite predecessors, that, in our opinion, have a rather controversial interpretation. Therefore, we introduced the concept of the *formal* and real *dynamic* reversibility. Naturally, from formal standpoint, we can consider arbitrary permissible possibilities, while of the applied standpoints the *"instant"* transition from infinite configuration to a finite one and vice versa, in our opinion, does not allow sufficiently transparent interpretations. At the same time, the theoretical studies of the dynamics reversibility of classical ***CA***–models are difficult enough *(however, like many other rather important problems in this class of parallel dynamical systems),* so the use of computer modeling for this purpose proved to be extremely effective. Like other tasks, for the experimental study of the ***CAs*** dynamics, we used the appropriate procedures programmed in the *Maple* and *Mathematica* systems, for which a rather large collection of software tools was created. Many of results related to the *reversibility* of different types due to the nonconstructability both of the *block* and *finite* configurations in classical ***CAs***, along with a rather detailed discussion of general and special questions in this direction, can be found, for example, in our works [10,11,113,141-148,155-161,164,182-196].

Algorithmical aspects of the nonconstructability problem for classical cellular automata. The algorithmic solvability of the nonconstructability problem is one of key issues of the mathematical theory of ***CA***–models and a number of its important applications, primarily when using ***CAs*** of both conceptual and practical models of spatially distributed dynamical systems, of which real physical systems are of particular interest [113]. In general terms, the solvability of the nonconstructability problem of is reduced to the question: ***Is there an algorithm to determine whether a classical CA–model will possesses the nonconstructability of type NCF, NCF–1, NCF–2 and NCF–3?*** In its general formulation, this problem remains open for today, but there are answers to many more specific but equally important issues that are independent interest. The most complete solution to the problem we obtained in the case of classical ***1-CA*** models. First of all, with regard to block and finite configurations, the following main result takes place, having a number of rather important applications.

For an arbitrary classical 1–CA the determination problem of the type of nonconstructability (NCF, NCF–1, NCF–2, NCF–3, constructible) is algorithmically solvable. The determination problems of minimum size

of the IBMEC, the presence of MEC and γ–configurations in classical 1–CAs models are algorithmically solvable.

The methods used in the proof allow not only to constructively determine the type of an arbitrary block and finite configuration [151,155], but also to establish the structure of many of their direct predecessors, which in many cases is quite important. For the general ***d***-dimensional case ***(d≥1)***, the question of determining for a block configuration of a particular type *(constructible, **NCF, NCF–3)*** is algorithmically solvable, but it does not say anything about the solvability problem as a whole, that is, about the nonconstructability existence of the ***NCF (NCF–3)*** type for an arbitrary ***d–CA*** model ***(d≥2)***. This problem in the case ***d≥2*** is unsolvable.

One of known approaches to solving the solvability problem of existence in the classical ***CA***–models of this or that type of nonconstructability is to determine the upper limit for minimum sizes of ***IBMEC, γ***-configurations or the nonconstructible configuration of the type ***(NCF, NCF–1, NCF–2, NCF–3)***. In the case of classical ***1–CAs*** models, we did just that, and in this direction we obtained a number of results of a certain independent interest [113,141,161]. This question plays an important role in assessing the minimum size of ***γ***–configurations, studying a number of dynamical properties of classical ***CA***–models and in study of the nonconstructability problem in general. As part of the study of the solvability of the problem ща nonconstructability, we and many other authors studied a relationship between the minimum sizes of ***NCF*** and ***IBMEC*** in classical ***CA***-models. In addition, contrary to the efforts made in that direction, no satisfactory solution had been obtained. However, a number of our results obtained in this direction [113,141,155,183] have led to the following assumption:

For classic d–CAs models (d ≥ 2), in general, it is impossible to obtain satisfactory quantitative estimates for the minimum size of NCF type configurations as a function of minimum IBMEC size, and vice versa.

Using the unsolvability of the well–known *"domino"* problem, we proved the main result [10,11,113,141-147,183]:

The existence problems in classical d–CA (d ≥ 2) of nonconstructability of types NCF, NCF–1, NCF–2, NCF–3 as well as MEC, MEC–1 and γ–configurations are algorithmically unsolvable.

The first part of the statement was proved by *J. Kari* [167] on the basis of another approach. Finally, on problems such as nonconstructability and reversibility, the closeness of sets of finite and infinite configurations relative to the global transition functions, the solvability of a number of problems for classical ***d–CAs (d ≥ 1)*** as well as on issues associated with them we obtained many results, which can be found, in particular, in [3, 10,11,113,141-148,155-161,164,182-196] and in some others our works.

5.3. Extreme design capabilities of classical cellular automata

The *axiomatics* of classical ***CA***-models are determined by their four basic parameters, namely: *the dimension **d** of homogeneous space **Z;** the states alphabet **A** of each unit automaton, the neighborhood index **X** and local transition function* $\sigma^{(n)}$**.** Within the framework of axiomatics, the question of the design capabilities of classical ***CA***–models is of particular interest: ***How serious are capabilities of the classical CA–models (within their axiomatics) in terms of generation of the finite configurations by them?*** Based on their own interests and tastes, many researchers determine the maximum generating capabilities of ***CA***–models in different ways within their basic axiomatics. Meanwhile, today we do not have a single idea of the maximum generative capabilities of classical ***CA***–models and it is a rather subjective. In contrast to nonconstructability, it is of considerable interest to determine the properties that reflect the maximum constructive properties of ***CA***–models with respect to generating by them of the finite configurations. Let's consider the most famous approaches based on the universal and self-reproducing finite configurations [3,141,156-161,164].

Universal finite configurations in classical cellular automata*.** In well–known monograph [168], *S. Ulam* formulated a rather interesting problem about existence of a simple universal matrix system. Its positive solution would give an interesting example of a simple generating formal system which can be sufficiently effectively investigated by known mathematical methods. We will need a number of necessary concepts and definitions as used below. *A square matrix **U(n, a)** of order **n** with members from a set **A** =* {0,1, ..., a–1***} *is called the universal matrix relative to the class of all matrices of order **m<n** if for each matrix **B(k,a)** (**k≤m**) there is an integer **j > 0** such that matrix **B** will be the main minor of the matrix* $U^j(n, a)$**.** So, within this definition, the following result solves the existence problem of a universal self–reproducing matrix system [10,11,113,141,170,183].

There is an integer $w_o > 0$ ***such that universal matrices U(n, a) cannot exist for arbitrary integers*** $n \geq w_o$ ***and*** $a \geq 2$***.***

It follows from this result that universal generating matrix systems of a sufficiently high order do not exist. Whereas for infinite matrices this question is still open, that is, in the original statement of *S. Ulam*, the problem of the existence of a universal reproducing matrix system is still waiting for its solution. Certain related materials can be found in [113].

As an interesting applied aspect of this problem, one can point out, for example, the use of classical ***CA*** to simulate logical deductive systems in pure mathematics. In this case, configurations from the set ***C(A, d, ϕ)*** are associated with logical calculus sentences, while an initial configuration of the ***CA***–model with its axiom and ***GTF*** with calculus production rules. Then the sequence of the global transition function *(**GTF**)* that applies to the initial configuration *(axiom)* is a proof *(conclusion)* in this deductive model. *Deductibility* and *completeness* problems are the main problems in such models. These two problems are directly related to the problems of the existence in classic ***d–CA*** *(**d*** $\geq$ ***1**)* models of configurations such as ***NCF*** and ***UFC*** *(universal finite configurations)*, respectively. The use of classical ***d–CA*** *(**d*** $\geq$ ***1**)* for modeling developing systems of cellular nature can be noted as the second applied aspect of the ***UFC*** existence problem.

The existence problem of the ***UFC*** for classical ***CA***–models, formulated by *S. Ulam* [169] for the case of regular lattices, is very closely related to the completeness problem of *H. Yamada* and *S. Amoroso* for the case of polygenic ***CA***–models [155]. This problem can be formulated as follows: ***Can there be a finite configuration or a finite set of them for a classical d–CA model (d*** $\geq$ ***1), of which the set C(A, d, ϕ) can be generated by the global transition function*** $\tau^{(n)}$ ***of the model?*** In other words, the question comes down to the permissibility of the following relationship, namely: $\cup_k <c_k>[\tau^{(n)}] = C(A, d, \phi)$; ***k=1..p.*** Consequently, the finite configurations $c_k \in C(A, d, \phi)$, that satisfy the above condition are called *universal finite configurations (denoted as **UFC**)*. For the case of finite ***CA***–models, the existence problem of the ***UFC*** has a positive solution, namely: ***There are finite d–CAs models (d*** $\geq$ ***1) that have one or all configurations as UFC*** [141]. A completely different picture takes place for the case of infinite classical ***CA***–models. Using results on the nonconstructability problem *(**NCF–1 & NCF**)*, we showed that such problem even in a more general setting has the negative solution for classical ***d–CAs*** *(**d*** $\geq$ ***1**)* models. So, the following result, having a number of applications along with many theoretical aspects, indicates this [10,11,113,141,143,183].

A classical d–CA model (d $\geq$ ***1) does not allow the presence of a finite set of universal finite configurations.***

The outline of the evidence, without breaking the commonality, is given in the assumption that there is such configuration $g_o \in C(A, d, \phi)$ that there is the relation $<g_o>[\tau^{(n)}] \equiv C(A, d, \phi)$, i.e. configuration g_o will be ***UFC:***

$$\tau^{(n)}: \ldots \rightarrow g_{-1} \rightarrow \boxed{g_0} \rightarrow g_1 \rightarrow g_2 \rightarrow \ldots \rightarrow g_j \rightarrow \ldots; \quad \cup g_j = C(A, d, \phi); \; j=0..\infty$$

But then there are only *four* possibilities for configuration predecessors g_{-1}: *(**1**) only finite configuration, (**2**) finite and infinite configurations, (**3**) only infinite configurations, (**4**) no predecessors*, i.e. g_o is ***NCF.*** So, cases *(**1,2**)* are obviously not allowed because otherwise a sequence $<g_o>[\tau^{(n)}]$

will be cyclic one, preventing the entire set of finite configurations from being obtained, by contraring to the assumption; the case *(4)* should also be excluded from consideration as invalid, since in this case each finite configuration having occurrence g_o will also be ***NCF,*** making it utterly impossible to generate the set $C(A, d, \phi)$ from the initial configuration g_o; finally, case *(3)* characterizes the configuration g_o as a nonconstructible one of ***NCF–1*** type, which ensures that the ***CA***–model has an infinite set of configurations of this type, which, it is easily to verify, also makes it impossible generating of the set $C(A, d, \phi)$ from the initial configuration g_o. Thus, the set $C(A, d, \phi)$ can`t be generated from a finite configuration by means of the global transition function of a classical ***CA***–model, i.e. for an arbitrary ***CA***–model there is no an ***UFC.***

In the same direction, under certain conditions, there is a stronger result expressed by the following suggestion [113,141,143,183]:

If classical model d–CA (d ≥ 1) has nonconstructability of the NCF type at existence for it of a set W of configurations of the NCF–1 type , then for the model there is no finite set of such configurations $c_g \in C(A, d, \phi)$ ***(g = 1..p) that the following determining relation occurs, namely:***

$$\bigcup_g \langle c_g \rangle \left[\tau^{(n)}\right] = C(a,d,\phi) \setminus W; \quad c_g \in C(a,d,\phi) \quad (g = 1..p)$$

Moreover, it directly follows from the above result that in some cases the narrowing of a set $C(A, d, \phi)$ of all finite configurations to the set of only constructible configurations that need to be generated does not lead to a positive solution of the ***UFC*** existence problem for classical ***CA***–models. Based on an algebraic approach using the results on nonconstructability, a more general and strong result [141] has been proved, which answers a number of previously raised questions, being a rather significant part of apparatus for dynamics studying of the classical ***CA***–models [113]:

If a classical model d–CA (d ≥ 1) with alphabet A = {0,1,2,3,4, ..., t–1}, where t – a prime number and $\tau^{(n)}$ ***– its global transition function, has a set M of configurations of the NCF and/or NCF–1 type, then there are no finite sets of global transition functions*** $\tau^{(n_j)}$ ***and configurations*** $c_j \in C(A, d, \phi)$ ***given in the alphabet A, for which two relationships exist:***

$$1)\ \bigcup_j \langle c_j \rangle \left[\tau^{(n_j)}\right] = C(a,d,\phi) \setminus M; \quad 2)\ \bigcup_j \langle c_j \rangle \left[\tau^{(n_j)}\right] = M; \quad (d \ge 1; j = 1..p)$$

At the same time, for the alphabet A (t is a composite number), there is a formulation of the result only with the relationship (2); this statement takes place for prime t and nonconstructability of NCF–2 type; that is, global transition function $\tau^{(n)}$ ***conditiones a set M of nonconstructible***

configurations of the NCF–2 type. If model d–CA (d ≥ 1) does not have nonconstructability of the NCF type, then the intersection of the sets of finite configurations generated by its global transition function from 2 different finite configurations is empty.

From this result follows a number of important properties of the classical ***CA***-models; in particular, it follows from it that the classical ***d–CA (d ≥ 1)*** models are not finitely axiomatized parallel formal systems, even if the nonconstructible finite configurations are excluded of the set ***C(A, d, ϕ).*** So, each set of nonconstructible configurations ***(NCF, NCF–1, NCF–2, NCF–3)*** relative to the completeness problem in classical ***d–CAs (d ≥ 1)*** has the same immunity as the set ***C(A, d, ϕ)*** itself. This result allows a much deeper understanding of the nonconstructability problem in the classical ***CA***–models. At the same time, it turned out that the exclusion of each of the four permissible nonconstructability types does not seriously affect the maximum constructive capabilities, in particular, in light of the existence of ***UFC*** sets for classical ***CA***–models. To a certain extent, this is a direct way to define the *complexity* concept of finite configurations in classical ***CA***–models.

We considered the existence of universal configurations in classical ***CA***–models at reducing of the requirement for their definition [147,155,161]: ***Are there classical d-CA (d≥1) together with initial finite configurations such that together they will generate all set of block configurations?*** In such setting, we do not require of generating of all finite configurations from some initial configuration, but only entering of set of all block finite configurations into the generated configurations, which is a significantly weaker condition in terms of the classical ***CA***–models concept. Thus, it is easy to convince that if a model ***d-CA (d ≥ 1)*** exists with this property, all configurations generated from some finite configuration and containing all finite block configurations will be reproducing in the *Moore* sense. While the nonconstructability of ***NCF*** type *(possibly **NCF–3**)* for model will be absent at presence in it the nonconstructability of ***NCF–1*** type. Obviously, if from a configuration ***c*∈C(A, d, ϕ)*** it would be possible to generate the whole set ***C(A, d, ϕ)*** by a global transition function, then ***c**** would be nonconstructible configuration of the ***NCF–1*** type.

To computer study the dynamic properties of classical models ***1–CAs*** we have created a number of tools in various programming systems; many of them programmed in *Maple* are presented in [98,109,111,115,171]. As many computer experiments show, these models have not only universal reproducibility in the *Moore* sense of finite configurations along with a non–zero finite configuration from which a sequence of configurations that will collectively contain all binary block configurations is generated.

Numerous computer experiments [98,113] allow us to formulate a rather interesting assumption, namely:

For each A =* {*0, 1, ..., p–1*} *(p is a prime), there is at least one classical model d–CA (d≥1) with the states alphabet A and the Neumann–Moore neighborhood index which from set of finite configurations of the form* □*w*□ *(w∈A*{*0*}*) generates in aggregate all finite block configurations.

The obvious contradiction between *block* and *finite* configurations from the set ***C(A, d, ϕ)*** is shown [141]: *If the set of block configurations in the aggregate can be generated by a finite set of finite configurations, while for finite configurations there is no finite set of finite configurations that generate in the aggregate the set* ***C(A, d, ϕ).*** So, this contradiction is one of the fundamental differences between *finite* and *block* configurations that largely determine their essence. Our results on the above–mentioned subject are available in more detail in [113,141,161,182-196].

Self–reproducing finite configurations in classical CA–models. If the existence problem of ***UFC*** characterizes the generating capabilities of classical ***CA***–models relative to a set of finite configurations in general, then universal reproducibility combines this possibility with structurally dynamic aspect of generating of configurations sequences in ***CA***–models. The essence of universal reproducibility is that any finite configuration in a classical ***CA***-model is self-reproducible in the *Moore* sense. At that, our research in this direction make it possible to establish many of interesting relationships between nonconstructability and universal reproducibility in the environment of classical ***CA***–models, as well as to solve a number of interesting enough problems of mathematical nature. In the future, by the self–reproducibility of a finite configuration ***t*** in the *Moore* sense we will mean the possibility of generating from it configurations containing any predetermined finite number of finite *block* configurations □***t***□. At that, it has been shown that ***if a classical d–CA model (d ≥ 1) generates a sequence of finite configurations containing in the aggregate all finite block configurations from a finite configuration c*, then c* will either NCF–1 or its finite configurations-predecessors other than c* will also have the same property of reproducibility.*** We have shown [141–147] that classical ***d–CAs*** models can have sets of complex enough finite self–reproducing configurations both in the absence and in the presence of ***NCF–1, NCF–2, NCF, NCF–3*** nonconstructability. The detected class ***L*** of linear classical models having the property of universal reproducibility of the finite configurations *(any finite configuration is self–reproducible in the Moore sense)* is most interesting in this regard. The classical model ***d–CA (d ≥ 1)*** is called a *linear* ***CA***–model if its local transition function $\sigma^{(n)}$ is defined by the following formula, namely:

$$\sigma^{(n)}(x_1,\dots,x_n) = \sum_{k=1}^{n} b_k x_k \ (mod\ a); \quad a, b_k - primes; \ x_k, b_k \in A = \{0,1,\dots,a-1\}, \ b_k \neq 0 \quad (k = 1..n)$$

Thanks to the works of a number of researchers, we can imagine a rather interesting result [113,155-161,164,182-196].

In a classical linear model d-CA ($d \geq 1$) each configuration $c \in C(A, d, \phi)$ is self-reproducible in the Moore sense, i.e., such model will possess the property of universal reproducibility of finite block configurations.

Using a number of our results from [141-147], we were able not only to significantly simplify the proof of this result, but also to a certain extent to characterize a whole class of similar ***CA***–models, hereinafter called *linear classical* ***d–CA****s* models *(**d ≥ 1**)*. Below, we believe that for models of this class, whose alphabet $A=\{0,1,...,a-1\}$ satisfies the condition $a = p^k$, where ***p*** and ***k*** are primes, the following result occurs:

Classical model d–CA ($d \geq 1$) with local transition function $\sigma^{(n)}$ defined as follows:

$$\sigma^{(n)}(x_1, x_2, \dots, x_n) = \left(\sum_{k=1}^{n} b_k x_k \right)^{a^m} \ (mod\ \ a)$$

(there are at least a pair of different integers j, $p \in \{1,2, ..., n\}$ such that b_j, $b_p \neq 0$; m is an integer or m = 1) possesses the property of universal reproducibility in the Moore sense of finite block configurations, where $a = p^t$ (p is a prime number; t, b_j and m are primes or m = 1), b_j, $x_j \in A = \{0,1,2, ..., a-1\}$; $j = 1..n$.

We discussed the question of self–reproduction in classical *linear* models ***d–CA (d≥1)*** and in some of their modifications rather in detail [113,141]. In this connection, the question arises: *Whether there are other classes of* ***d-CA*** *models* ***(d≥1)*** *that have universal reproducibility of configurations, and how could they be described formally**?*** In this direction, it is shown [113,141,161] that classical ***1–CA*** models with local transition functions defined in a special way do not have nonconstructability of the ***NCF*** type whereas there is nonconstructability of the ***NCF–1*** type for them. At that, the number of such ***1–CA*** models with states alphabet $A = \{0,1,2,\dots,a-1\}$ and neighboorhood index $X = \{0,1,2,\dots,n-1\}$ is equal to $\frac{(a-1)^{a-1}}{a^a}(a!)^{a^{n-1}}$. As it turned out, among such large number of models, many of models were found that are different from linear ones, but possess the property of self-reproducibility in the *Moore* sense. Meanwhile, the general criterion for the presence of self-reproducibility in a classical ***CA***–model in the *Moore* sense is unknown to us for today. Based on many computer experiments and a number of our theoretical results on nonconstructability in classical ***d–CA*** models *(**d ≥ 1**)* and on the dynamical properties of the models, we formulated the following a rather interesting assumption, namely:

The nonconstructability existence of the NCF–1 type in classical d–CA ($d \geq 1$) models in the absence of nonconstructability of the NCF type in these models is necessary, however not enough for the existence of self-reproducing block configurations in the Moore sense.

In a certain sense, this assumption could serve as a *filter* when checking ***CA***–models for their self–reproducibility property in the *Moore* sense. Our theoretical studies, based on certain dynamic properties of classical ***1-CA*** models associated with nonconstructability of ***NCF-1*** type together with numerous computer studies, allowed us to formulate the proposal:

In a classical 1–CA model with the states alphabet A = {0,1,2,3,...,a–1}, neighborhood index X = {0,1,2,...,n–1} and local transition function:

$$\sigma^{(n)}(x_1, x_2, ..., x_n) = \sum_{k=1}^{n} x_1 \quad (mod\ \ a) \qquad x_1 \in A;\ k = 1..n$$

where a is represented in form $a = p_1^{t1} p_2^{t2} \cdots p_g^{tg};\ p_j, t_j - primes;\ j = 1..g,$ ***each finite block configuration is self–reproducing in the Moore sense.***

Note, the rate of generation of the required number of copies of a block configuration in such *linear* ***CA****s* models is significantly lower than if ***a*** are prime numbers. It is therefore only natural to draw the conclusion on the basis of this proposal:

In a strictly linear classical model 1–CA with alphabet A = {0,1,...,a–1} (a∈{2,3,...,10}) and arbitrary neighborhood pattern, any configuration of a finite block determined in the alphabet A is self–reproducing in the Moore sense.

In particular, a number of procedures programmed in *Mathematica* can be used to computer study this problem. Meanwhile, it should be borne in mind that due to insufficiently efficient cyclic expression processing algorithms, the *Maple* is the more preferred system than *Mathematica* for many tasks of computer research of ***CA***-models dynamics [113,155-161]. In this context, we considered [161] in a certain sense some generalized class of linear classical ***CA***–models characterized by dynamic property of universal reproducibility in the *Moore* sense. A set of similar *linear* ***CA***–models forms a semigroup relative to the operation of composition; at the same time, saving the self–reproducibility property in the *Moore* sense.

At the same time, there are nonlinear classical ***1–CA****s* models for which the finite configuration and the inverse to it are self–reproducing in the *Moore* sense. Such models have nonconstructability of the ***NCF–1*** type without nonconstructability of ***NCF*** type, and the generation of copies of both direct and reverse finite configuration is carried out simultaneously. Interesting enough examples of classical ***1–CA****s* models of this type were obtained and a group of classical ***1–CA****s* models with self–reproducibility in the *Moore* sense of finite configurations and inverse to them have been established. At the same time, the ***1–CA*** models of this group differ from linear classical models [113,141-147,161]. During computer study, many interesting results were obtained regarding self-reproducing in the *Moore* sense configurations in the class of linear classical ***CA***–models ***(d = 1, 2)***. The problem of decomposition of global transition functions in classical ***CA***-models is discussed below; we used this approach to create nonlinear classical ***1–CAs*** models that have universal reproducibility in the *Moore* sense. Interesting examples of such models can be found in [141,161]. So, numerous experiments with a procedure programmed in the *Mathematica* made it possible to formulate a rather convincing proposal, namely:

A composition of the global transition functions of binary models 1–CA with qualification numbers from the set {6, 60, 90, 102, 105, 106, 120} have the property of universal reproducibility in the Moore sense.

Along with studies of classical ***CA***–models with the property of universal reproducibility, it is interesting to identify other classes of ***CA***–models with a certain general property, interesting from both theoretical and the applied points of view, and effectively characterize these classes in terms of new or previously studied concepts and categories. Given the question, the research of the class of ***CA***–models with symmetric local transition functions turned out to be quite interesting. The analysis of classical ***d–CAs*** models ***(d = 1,2)*** from this class on the basis of both theoretical and computer studies [147] made it possible to formulate a rather interesting proposal that in a subclass of this class there is an infinite set of models having the property of universal or essential reproducibility in the *Moore* sense. So, a rather interesting proposal seems quite convincing to us:

Among the classic models d–CA (d ≥ 1) with symmetric local transition functions in the presence of NCF-1 nonconstructability and not having NCF nonconstructability, there are infinitely many models which have universal or essential reproducibility in the Moore sense.

Moreover, it is shown that the following result occurs [113,146,147]:

The class of d-CA models (d≥1) relative to the universal reproducibility property in the Moore sense is wider than the CA–models class that are defined by linear local transition functions and their superpositions.

Note that the modeling method using the software created in the *Maple* and *Mathematica* systems, allowed to determine a number of types of the classical ***1–CA*** models that have the property of essential reproducibility of finite configurations along with some other rather interesting dynamic properties of ***CA***–models of this class [113,161]. Analysis of the above results, together with a number of theoretical considerations, allows us to conclude that there is no linearity of classical ***CAs*** models as a root cause that forms the basis of universal or essential reproducibility in the *Moore* sense. More precisely, both universal and essential reproducibility in the *Moore* sense has deeper roots, and their thorough identification is of the undeniable interest. In any case, based on the results we obtained, quite interesting examples of classical ***CA***–models are obtained, which have universal reproducibility and which are different from both the class of linear classical models and the wide class of ***CA***–models formed through composition of their global transition functions, and having the property of universal reproducibility in the *Moore* sense. Generally speaking, self-reproducibility in classical ***CA***–models is studied with respect to finite configurations, but this phenomenon can also be generalized to the case of infinite configurations [10,11,113,141,161,182-196].

As part of the study of linear classical ***CA****-models* and their modifications *(in particular, formed on the basis of the composition of global transition functions),* it became advisable to clarify the effect of the *symmetry* of the local transition functions on reproducibility in the *Moore* sense in classic ***CA***–models. To this end, we identified a group of classical ***1–CA*** models with *symmetric* local transition functions and tried to elicit relationship with reproducibility in the *Moore* sense. Theoretically and based on wide computer analysis, it is shown [141,161] that models of this group have the property of universal or essential reproducibility in the *Moore* sense and have interesting dynamics of generating copies of self–reproducible finite configurations. Thus, it is only natural to assume that the universal reproducibility property in the *Moore* sense in classical ***CA***–models is primarily based on some form of complete or essential symmetry of local transition functions relative to the main diagonals of structured–designed substitution rules that determine local transition functions, along with the absence of the ***NCF*** nonconstructability in the presence of the ***NCF–1*** noconstructability, but not their *linearity*. At that, for the states alphabet $A = \{0,1,...,a\}$ at $(a + 1) = p^h$, where p, h are primes or $h = 1$, the speed of generating copies of the initial configurations of the same length, usually depends on the type of *symmetry* of the local transition functions [161].

The general scheme ***W*** for organizing *symmetric* local transition function for a classical ***1–CA*** model can be found in [161]. Scheme ***W*** presents two options for *symmetry* of local transition functions with respect to the two main diagonals of parallel substitution subblocks, which constitute a common block of ordered parallel substitutions that determine the local transition function of the ***1–CA*** model. The ***1–CA*** model thus defined has ***NCF–1*** nonconstructability in the absence of ***NCF*** nonconstructability. Obviously, the number of different classical models with local transition functions defined by the schemes ***W*** is $2(a–1)!$. Theoretically, and based on a sufficiently extensive computer analysis, it is shown [113,141,161, 182-196] that the following proposal can be formulated:

Classical 1–CAs, whose local transition functions are determined using relationships of W type, will have essential or universal reproducibility in the Moore sense*. *This result can probably be generalized to the case of the d–CAs models (d > 1).

The generation capabilities of classical ***CA***–models include the problem of the existence of *periodical* configurations. It is shown [141] that if in a classical ***CA***–model there are *periodical* configurations with a minimum ***p***–period, their number is infinite, and there are periodical configurations of infinitely large size with the same ***p***–period. If periodic configurations exist in a ***CA***–model with minimum periods ***p*** and ***q*** ***(p ≠ q)***, then at least periodical configurations exist in it with minimum period ***gs = gs(p, q) = LCM(p, q),*** where ***LCM*** is least common multiple ***p*** and ***q.*** This raises two main issues: *(**1**) obtaining upper estimates for the size of minimum periods as a function of main parameters of a **CA**–model as well as (**2**) elucidating the algorithmic solvability of problem of having periodical configurations in a classical **CA**–model, in addition to the trivial case of periodical zero configuration.* For classical ***CA***–models, the lower limit of the minimum period size is set, expressed by the next result:

There are classical d–CAs (d ≥ 1) models with the Moore neighborhood index that possess the periodical finite configurations with a minimum period p ≥ 2|gs| – 2, where |gs| is the diameter of a gs configuration.

It is proven [141] existence in classical ***1–CAs*** of *periodic* configurations of relatively small size with a fantastically large minimum period size. A positive solution of the above first question entails the solvability of the second question, while the unsolvability of the second, in turn, entails a negative solution of the first. For today, both questions remain open even for the case of classical ***1–CAs*** models.

Cellular automata on partition ***(CAoP)***, defined above, are of particular interest for physical modeling, allowing for fairly simple programming of the *reversibility* dynamics. As for the issues of existence of universal configurations ***(UFC)*** in ***CAoP***–models, then the following result takes place [113,141,147]: ***Model d–CAoP (d ≥ 1) cannot possess a finite set of UFC.*** An interesting enough picture occurs regarding the existence of self–reproducing configurations in the *Moore* sense in ***CAoP***–models. It is shown that the ***d–CAoP*** models ***(d ≥ 1)*** can have *essential* or *universal* reproducibility in the *Moore* sense, and their dynamics will be *reversible*. However, as in the case of classical ***CA***–models, the following negative result occurs: ***There is no 1–CAoP model that can double an arbitrary finite configuration defined in the same model states alphabet.*** Result has interesting enough independent appendices [10,11,113,141,183].

5.4. The complexity problem of finite configurations in classical cellular automata

Complexity in all its commonality is one of the vaguest and intriguing concepts of modern natural science. In our opinion, in many ways the main reason for this is the intuitive essence of the concept. At the same time, we emphasize that the most fundamental problem of development is understanding of how the system can itself become more complicated, and how complex the original system should be for this purpose. One of the difficulties in solving this problem, grandiose in many aspects, is the lack of a satisfactory measure of complexity. At that, it is possible that for the general complexity concept there is simply no a single approach, despite the fact that numerous serious attempts have been made in this direction, while the very concept of complexity is essentially multifaceted and defined by the sphere of its application. So, studies of the *complexity* concept are extremely desirable and are carried out in various areas.

For formal modeling of various discrete processes and phenomena in the classical ***CA***–models, particular interest is associated with dynamics of the finite configurations. Indeed, modeling of some process is presented by the dynamics of a classical ***CA***–model *(i.e. the appropriate history of the initial finite configurations in it)*. In this context, the question arises about the complexity of finite configurations that form the history of the development of a certain process or object simulated in the classical ***CA***–model. Now, three main approaches to the concept definition of "*quantity of information*" are known, that are associated with complexity concept of finite objects, namely: *combinatorial*, *probabilistic* and *algorithmical*, based on the theory of recursive functions and abstract automata. So, for example, within the framework of algorithmic approach, *A. Kolmogorov* determined the relative complexity of a certain object ***A*** relative to object ***B*** by means of the minimum length of a program for obtaining the finite object ***A*** from the finite object ***B.*** At the same time, *A. Kolmogorov* chose their binary numbers in a certain formal numbering as representatives of these objects, while as the output program – the program of work of the corresponding *Turing* machine.

The approach we have proposed to determine the complexity of finite configurations based on ***CA***–axiomatics is, by its essence, algorithmical too, but differs from the approach of *A.N. Kolmogorov*. The essence of our approach to definition of a *complexity* concept of finite configurations consists in assessment of complexity of generation of an arbitrary finite configuration from a *primitive* configuration $c_p \in C(A, d, \phi)$ *(for example,* c_p = □***1***□ *for **1–CA**s)* by means of finite number of the global transition functions $\tau^{(n_k)}$ from some fixed set G_f of functions which we will call a *basic* set. To rigorously define the *complexity* concept, we will need some fundamental results related to the dynamics of finite configurations in the classical and polygenic ***CA***–models. Nonconstructability problem occurs for *monogenic* and *polygenic* ***CA***-models. In the *second* case, the problem is known as the *completeness* problem and is defined as: ***can an arbitrary finite configuration be generated from some primitive configuration by a finite sequence of global transition functions of polygenic CA-model?*** This problem attracted the attention of many researchers, who received a number of quite interesting results in this direction, while the important result of *M. Kimura* and *A. Maruoka* completely completes the solution to the *completeness* problem [10,11,113,155,164,183].

An arbitrary d–dimensional nonzero configuration $c \in C(A, d, \phi)$ can be generated from a primitive configuration $c_p \in C(A, d, \phi)$ with the help of some finite sequence of global transition functions $\tau^{(n_k)}$ of a polygenic model d–CA ($d \geq 1$).

So, the *completeness* problem to a certain extent characterizes the design capabilities of polygenic ***CA***–models, proving rather wide capabilities of this class of ***CA***–models regarding the generation of finite configurations. Meantime, from the above results of *M. Kimura* and *A. Maruoka* directly follow the following a rather interesting result:

An arbitrary d–dimensional configuration $c \in C(A, d, \phi)$ for a polygenic model d–CA ($d \geq 1$) can be generated from a certain initial primitive configuration $c_p \in C(A, d, \phi)$ with the help of a certain finite sequence of d–dimensional global transition functions $\tau^{(n_k)}$ of a fixed (basic) set G_f.

This result is of both theoretical and applied interest, for example, in the systems for processing and storing graphic information of various types *(for example, in image databases)*, as well as in different systems for the encoding and decoding information. On the other hand, it should be noted that there is a certain result directly related to our results on the existence problem of the ***UFC*** for classic ***CA***–models [113,147,154,161,164,182].

There are no finite sets of d–dimensional c_k configurations from the set $C(A, d, \phi)$ along with global transition functions $\tau^{(n_k)}$, which are defined in the same finite alphabet A, that satisfy the relation, namely:

$$\bigcup_k <c_k>\left[\tau_k^{(n_k)}\right] \equiv C(A,d,\phi) \qquad (A=\{0,1,\ldots,a-1\};\ d \geq 1;\ k=1..p)$$

We have presented several options for proof of this result, which can be found, in particular, in [113,141]. So, the above results provide a fairly strong basis for a rigorous justification of our concept of the complexity of finite configurations based on ***CA***–axiomatics along with a number of other results in this direction. In particular: *Even polygenic **CA**–models are not finitely axiomatizable formal systems, that is, it is impossible for them to determine a finite set of configurations (axioms) from which it would be possible to derive the entire set **C(A, d, ϕ)** by means of a finite set of global transition functions (derive rules)*. Let us now turn to the definition of the *complexity* concept of finite configurations, which is of certain theoretical and gnoselogic interest.

Let G_f be a finite set of ***d***-dimensional global transition functions given in some finite alphabet ***A,*** by which ones, during finite number of steps, an arbitrary finite configuration w^* can be generated from some primitive configuration $c_p \in C(A, d, \phi)$, i.e., the following rules exist for outputting finite configurations from some primitive configuration c_p:

$$w^* = c_p \tau_1^{m_1} \tau_2^{m_2} \tau_3^{m_3} \cdots \tau_n^{m_n} \qquad \left(\tau_k \in G_f;\ \tau_j \neq \tau_{j+1};\ k = 1..n;\ j = 1..n-1\right)$$

where m_k – the applying multiplicity of global transition functions $\tau_k \in G_f$ $(k = 1..n)$. Let's say that a configuration $w^* \in C(A, d, \phi)$ is generated from a certain simplest configuration $c_p \in C(A, d, \phi)$ in at least $r = \sum_k m_k$ steps of global transition functions $\tau_k \in G_f$ $(k = 1..n)$. So, for classical models ***1-CA*** the configuration c_p = □***1***□ can be selected as the simplest configuration. In addition, two arbitrary finite configurations $\tau_i, \tau_j \in G_f$ are assumed to be different $(\tau_i \neq \tau_j)$ only if there is a relation $(\exists c \in C(A, d))(c\tau_i \neq c\tau_j)$.

If in the above generating chain there are $(n–1)$ pairs of different global transition functions $<\tau_i, \tau_j>$ $(j = 1..n–1)$, then we will say that in the above generating chain of configurations $w^* \in C(A, d, \phi)$ from the configuration $c_p \in C(A, d, \phi)$ will exist $(n–1)$ levels L_k, that are defined by the following binary *signaling* function, namely: $L_k = If(\tau_k \neq \tau_{k+1}, 1, 0)$ $(k=1..n–1)$. In [161] is diagram *(Fig. 10)* illustrating the described process of generating *(optimal output strategy)* of an arbitrary finite configuration $w^* \in C(A,d,\phi)$ from a simplest finite configuration c_p according to the above outputting chain. It should be noted that this diagram can serve as a good illustration for a number of recearches related to the complexity concept of the finite configurations in classical ***CA***–models. Because of this, the complexity of an arbitrary finite ***w*** configuration can be determined as follows:

The complexity of an arbitrary configuration $w \in C(A, d, \phi)$ $(d \geq 1)$ ***based on CA–axiomatics is calculated using the generalized formula:***

$$SL(w) = \min_{\tau_k \in G_f} \prod_{k=1}^{n-1} p_k^{m_k}$$

where p_k ***is the k-th prime number, and*** m_k ***are determined based on the***

above inference chain of the finite configurations of a d–CA polygenic model (d ≥ 1).

Based on this definition, a number of fairly important properties of finite configurations were obtained in *classical* and *polygenic* ***d–CA*** models, which characterize them relative to the introduced concept of complexity [113,141-147]. Certain results in this direction have a number of rather interesting applications in theoretical and applied aspects. Among them, in particular, we note the following a rather interesting result:

For an arbitrary integer d ≥ 1, the set C(A, d, ϕ) of finite configurations of d-dimension contains configurations of given complexity relative to the finite base set G_f of global transition functions defined in a certain finite alphabet A of a polygenic model d–CA (d ≥ 1). In dimension d ≥ 1, there are global transition functions $\tau \notin G_f$, generating from a given gs configuration $gs \in C(A, d, \phi)$ of limited complexity, a configuration of any pre–defined complexity in the sense of the above–stated complexity definition.

This result states that if the global transition functions constituting the base set G_f generate configurations of only limited complexity, then the configurations of any complexity can be created by the global transition functions not belonging to the set G_f. This result gave rise to many rather interesting questions, one of which is the question of the number of finite configurations of the same complexity with respect to the given base set G_f. The following result provides an opportunity to clarify this issue to a great extent:

There is an infinite number of basis sets G_f of d–dimensional global transition functions defined in a finite alphabet A, with respect to each of which there are infinite sets S_j of finite configurations of the same complexity in the sense of the above–stated complexity definition.

This result allows us to solve a number of interesting enough questions formulated in our works [3,113,141,147]. A fairly detailed study of the basic set G_f used in determining the complexity concept of the finite configurations in classical ***CA***–models, as well as the properties of global transition functions which form a set G_f, allows us to very significantly clarify not only new properties of the introduced complexity concept, but also provide an effective apparatus for research of the dynamics of ***CA***–models such as *classical*, *polygenic* and *non–deterministic* in some cases. So, in particular, it is very important to study the minimum base set G_f, containing the smallest number of the global transition functions $\tau_k^{(n_k)}$. Studying the completeness problem in *polygenic* models, *M. Kimura* and

A. Maruoka presented constructive evidence; however, they did not use an optimizing technique. In general, a detailed study of the base G_f sets of global transition functions is still absent, while a number of interesting results have been obtained with respect to the narrower class of binary ***1–CAs*** models *(see* [3,113,141,147,161,164] *and works cited in them)*.

There is a minimum basic set G_f of four binary 1–dimensional global transition functions $\tau_k^{(n_k)}$; at least one function of them possesses the NCF–1 nonconstructability. Moreover, with respect to such set G_f, there are infinite sets of finite configurations of the same complexity.

The method of proving [141] this result proved to be useful in obtaining a number of results that have a number of important applications in the study of the dynamics of classical ***CA***–models, both by allowing to get answers to a number of questions and slightly more deeply revealing the essence of the introduced concept of complexity of finite configurations based on ***CA***–axiomatics. In this regard, it should be noted, the concept of complexity of some algorithm largely depends on both the algorithm itself and its specific implementation. For today, there is no traditional, more accurate definition. So, the results regarding the assessment of complexity for algorithms may well be of a significantly different nature. For example, the complexity of the normal *Markov* algorithm is defined by the length of the recording of all its substitution formulas, while the complexity of the *Turing* machine is usually determined by the product of the number of states of finite automaton and alphabet symbols of the external tape. At that, the complexity of an algorithm implemented in the ***d–CA*** model $(d \geq 1)$, it is natural to define by the formula $W = d*a*n*p$, where d – dimension of the model, a – cardinality of its alphabet, n – size of the neighborhood index and p – minimum number of parallel rules that define the local transition function required for the realization algorithm. So, the above complexity concept of finite configurations significantly affects the comparative characteristics of various algorithms. Therefore, the conceptual basis of the compared formal algorithms needs to be given much more attention [10,11,113,141,161,182-196].

In researches the complexity problem of finite configurations, we rather substantially used the concept of the minimum basic set G_f and some of the dynamic properties of the global transition functions composed it. In this direction, in particular, the properties of similar minimal bases were studied in more detail, given their importance for the research of deeper properties of the dynamics of the classical ***CA***–models. For the sake of simplicity, we will limit ourselves to the case of binary polygenic ***1–CA*** model, the set $C(B, 1, \phi)$ of finite binary configurations, and binary global transition functions $\tau_k^{(n_k)}$, whose local transition functions are determined by the following parallel substitutions, namely:

		(a)	(b)	(c)
000	→	*0*	*0*	*0*
001	→	*1*	*1*	*1*
010	→	*0*	*1*	*0*
011	→	*1*	*0*	*0*
100	→	*0*	*1*	*1*
101	→	*1*	*0*	*1*
110	→	*1*	*1*	*0*
111	→	*0*	*0*	*0*

		(d)
00	→	*0*
01	→	*1*
10	→	*1*
11	→	*1*

In view of the assumptions made, there is the following important result characterizing the global transition functions for the minimum base set G_f for ***1***–dimensional case of binary polygenic ***CA***–models [141,161].

The minimum basis set G_f includes four 1–dimensional binary global transition functions $\tau_k^{(n_k)}$, whose local transition functions $\sigma_k^{(n_k)}$ are defined by the above parallel substitutions; at that, the global transition functions constituting the basis set G_f have finite configurations of the types according to the table below.

LTF\NCF	*NCF*	*NCF–1*	*NCF–2*	*NCF–3*	*ACCF*
(a)	–	+	–	–	+
(b)	–	+	–	–	+
(c)	+	+	–	–	+
(d)	+	–	+	+	–

The minimum basis set G_f for 1–dimensional non–binary case consists of global transition functions $\tau_k^{(n_k)}$, which possess configurations of the types such as NCF and/or NCF–1, NCF–2, and possibly NCF–3 along with absolutely constructible configurations (ACCF).

The issue is discussed in some detail in [141] together with consideration of the existence of unconstructability types for global transition functions constituting the minimum set G_f of global transition functions. The above result achieved some problems from [172]; the result can be used to study the complexity problem of ***1***–dimensional finite configurations in a finite alphabet. In particular, based on this result, we can receive the simplest justification for the complexity concept of finite configurations that was introduced by us for the ***1***–dimensional binary case. In addition, a result of such justification takes the following form, which is of independent interest as part of the apparatus for studying classical ***CA***–models.

Any 1–dimensional binary configuration $c\in C(B, 1, \phi)$ is monotonically generated from the primitive configuration c_p = □1□ using the global transition functions $\tau_{jk}^{(n_k)}$ from a fixed finite set G. At the same time, there is no such finite system $\{c_k, \tau_{jk}^{(n_k)}\}$ that the following determining relation will occur, namely:

$$\bigcup_k \langle c_k \rangle \left[\tau_{j_k}^{(n_k)} \right] = c(B,1,\phi);\ c_k \in c(B,1,\phi)\ \ (n_k \in \{2,3\};\ j_k \in \{0,1,2,3\};\ k=1..p)$$

As the base set G_f, we can choose a set G of the binary global transition functions, relative to which the complexity concept of one–dimensional binary finite configurations in classical CA–models is determined.

It should be noted that this result is generalized to the case of any finite alphabet ***A*** of the states of an elementary automata of an arbitrary model ***1–CA.*** Moreover, based on this result, it is possible to obtain the simpler evidence *(in many cases constructive)* of previous results, along with a number of other interesting results that concern the complexity of finite configurations for the case of binary ***1–CA*** models [141-147]. These and related questions are discussed in some detail in [141,161]. Meanwhile, the complexity problem of finite configurations in classical ***CA***–models, despite our results and other authors' results, has a number of open issues along with promising directions for further researches requiring solutions from various standpoints *(see references in* [161]). Let us briefly discuss two different approaches to defining the complexity concept of the finite configurations in classical ***CA***–models, namely: *configuration* and *block* approaches, the essence of which is the following.

First of all, the complexity of a finite configuration means the possibility of ***CA***–model or finite set of similar models to generate the set ***C(A, d, ϕ)*** from one or a finite set of initial finite configurations. From the results obtained, it follows that in the arbitrary determination of the finite base set G_f, finite configurations of predetermined complexity will still exist in the set ***C(A, d, ϕ)*** of all finite ***d***–dimensional configurations defined in the arbitrary finite alphabet ***A.***

A completely different picture occurs at definition of *block* complexity *(a wider generation possibility)* when instead of the finite configurations ***c*** = □$hx_1x_2 ... x_nh$□ [□ *– zero configuration of an infinite number of symbols* ***'0'***; $x_j\in A$, ***j = 1..n***; $h\in A\backslash\{0\}$] we will considere the block configurations, i.e. *block* configurations of the form $<x_1x_2 ... x_n>$ $\{x_j\in A, j=1..n\}$. At this approach, another situation is quite real. In particular, it is shown [141] that there are binary models ***1–CAs*** for which there are infinite sets of the finite *nonconstructible* configurations of ***NCF–1*** type which collectively generate the entire set ***C(B, 1, ϕ)*** of all finite configurations.

In the context of the above, there is a certain interest in the possibility of classical ***CAs*** to generate the same sequences of finite configurations. In this direction there is the following result [141,161,182-196], namely:

There are CA–models pairs generating the same sequence of the finite configurations for a certain finite set of initial finite configurations, while the existence problem of the d–CA (d ≥ 1) model generating an arbitrary sequence of configurations is, generally speaking, unsolvable.

Such questions are of undeniable interest due to the fact that the study of the dynamic properties of the classical ***CA***–models, as formal objects, is based on the study of the sequences of finite configurations generated by them. In this regard, the question arises about the intersection of sets of sequences generated by two classical ***CA***–models, which is unsolvable.

The complexity problem of finite configurations in classical ***CA***–models is of great importance not only in context of studying both certain formal deductive systems, but also in the case of embedding in them developing systems of cellular organization and certain their phenomena. Moreover, the given problem is most directly related to the problem of studying the complexity of self–organizing biological cellular systems, which is quite relevant for modern mathematical and developmental biology.

As is well known, cybernetic study on developmental biology still lacks a fairly satisfactory approach to assessing the complexity of developing biological systems. Our mathematical approach in this direction can be enough fruitful and promising. So, the results obtained along with our other results on the complexity problem of finite configurations in ***CA***–models will not only actually form the problematics and solve a number of its main problems in general, but also they allow to formulate many open questions and rather promising directions for further study, that are of significant independent interest in the theoretical and applied aspects of ***CA***–models problematics.

Our findings on the complexity of finite configurations in the context of ***CA***–axiomatics allow us to better clarify the essence of the complexity concept depending on the axiomatics used. So, in the axiomatics of the classical and polygenic ***CA***–models there are binary finite configurations of any given complexity, while in other axiomatics, for example, in the *A. Kolmogorov* axiomatics, all binary words printed on a *Turing* machine on an output tape will be only of limited complexity. Thus, most likely, there is no concept of some absolute complexity of finite objects along with the complexity concept as a whole; i.e., to a large extent, the concept of complexity is pronounced axiomatical in nature. Many results on the complexity problem in classical and polygenic ***CA***–models remain valid for ***CAoP***–models, having interesting physical applications [113,141].

5.5. Parallel formal grammars and languages defined by the classical cellular automata (*CA–models*)

The *theory of formal grammars (**TFG**)* is the central part to mathematical linguistics, providing formal resources for study of the functioning of the language. The ***TFG*** stands out against the background of other sections of mathematical linguistics with a greater complexity of the apparatus used, which is similar to the apparatus of algorithm theory and apparatus of the general automata theory, with which it has many points of contact and intersection. The mathematical significance of generating grammars is determined by the fact that they are one of the means of effectively determining important sets of words. At the same time, a class of formal languages generated using any grammars will coincide with the class of all recursively enumerated sets. Of this standpoint, the formal grammars of the *Chomsky* classical hierarchy are of particular interest. Therefore, the study of classes of abstract automata that are equivalent to classes of formal grammars describing the same formal languages is essential.

Since ***TFG*** is a part of the automata theory, the study of the dynamics of ***CA***–models from its standpoint undoubtedly deserves special attention, so a number of our works are devoted to these problems. Meanwhile, the theory of parallel formal grammars can be effectively used not only in creating the theory of parallel programming along with the architecture of computing systems of parallel action of new generations, but also in creating a linguistic basis for describing the dynamics of various space–distributed systems of cellular nature. To this end, to study the languages generated by classical ***CA***–models, in *1974* we defined a class of formal *parallel* grammars, the so–called $\boldsymbol{\tau_n}$*–grammars* [136-138]. At the same time, were generally researched $\boldsymbol{\tau_n}$*–grammars*, determined by *classical* and *nondeterministic* models ***1–CA****s*, however similar approach can be extended to the ***d-CA*** models ***(d ≥ 2)*** and some other types of ***CA***-models. With this approach, classical ***CA***–models can be considered as a subclass of *formal parallel grammars (**FPG**)* that do not use non-terminal symbols and whose output is carried out in an absolutely parallel way. Grammars of this type are similar to the known *Lindenmayer* systems ***(L****–system),* they can be quite successfully used for the formal linguistic description of the dynamics of different cellular objects and many parallel discrete processes and phenomena. At the conceptual level, we studied parallel $\boldsymbol{\tau_n}$–grammars in accordance with ***TFG*** traditions resulting in a number of ***FPG*** characteristics of this class that are useful from many standpoints.

Informally, $\boldsymbol{\tau_n}$*–grammars* are defined as follows. By analogy with basic concepts of ***TFG,*** the alphabet ***A*** of an elementary automaton of classical model ***1-CA*** is considered to be an alphabet of τ_n–grammar, and its local transition function $\sigma^{(n)}$ defines a set of parallel output rules. In grammar, the initial finite configuration of the model determines the *axiom*, and the finite configurations generated from this axiom are the language *words* determined by such parallel τ_n–grammar. Similarly to the usual formal grammar in a classical ***CA***–model, new configurations *(language words)* are output from an initial finite configuration c_o *(axiom)* by sequentially applying the local transition function $\sigma^{(n)}$ *(output rules)*. Whereas there are two important differences between traditional formal grammars and parallel τ_n–grammars, namely:

– *output rules in a* τ_n*–grammar are applied simultaneously in absolutely parallel way;*
– *terminal and non–terminal symbols do not differ from each other in the states alphabet* **A** *of the parallel* τ_n*–grammar.*

In ***TFG,*** some formal language is defined as the set of all terminal words generated from an axiom c_o by grammar output rules. Whereas the $L(\tau_n)$-language is defined as the set of all finite configurations *(words)* that are generated from initial configuration *(axiom)* by means of simultaneously applying parallel substitutions defined by the local transition function to all symbols of the current configuration *(word)*. We studied τ_n–grammars according to ***TFG*** traditions regarding a number of characteristics of this class of parallel grammars [113,136-138]. Unless otherwise indicated, the parallel τ_n*–grammars* and $L(\tau_n)$*–languages* below are considered which defined by classical models $<Z^1, A, \tau^{(n)}, X>$ with alphabet $A = \{0,1,...,a-1\}$ and neighbourhood index $X = \{0,1,...,n-1\}$.

Studying the closure property of a class of formal languages in relation to traditional operations in ***TFG*** is a classical approach to the mathematical characteristic of this class. There are two basical reasons for considering these operations regarding parallel $L(\tau_n)$–languages [136]. The following result determines the behavior of $L(\tau_n)$-languages in relation to traditional operations studied in the classical theory of formal grammars [113,183].

The class of languages $L(\tau_n)$ is not closed with respect to operations such as finite transformation, homomorphism, iteration, intersection union, product and addition, while the class of these parallel languages is closed with respect to the reverse operation.

Meanwhile, the following fact can be attributed to the most significant features of τ_n–grammars: *most approaches based on standard methods and apparatus of study in* ***TFG*** *do not apply to the class of* τ_n*-grammars,* assuming the use of new non–standard methods here. For example, the use of methods for the theory of recursive functions made it possible to solve a number of questions in the theory of τ_n–grammars [113,155]. In particular, it is shown that *the dictionary function defined by the parallel mapping* $\tau^{(n)}: C(A,\phi) \rightarrow C(A,\phi)$ *is primitive–recursive function*. Whereas there remain a number of issues related to the application of methods and results of recursive function theory to the study of the dynamic properties of classical ***CA***–models; at the same time, to date, this approach provides significant assistance in this direction [113,136-138,141,183].

Whereas for the case of parallel $L(\tau_n)$-languages, such approach allows to obtain quite interesting results, in particular, the following can be noted:

In general, there is no finite set of languages $L(\tau_n)$***, whose union forms the complement of a certain language of the same class; wherein the addition of a finite set of*** $L(\tau_n)$***–languages cannot be again a language of the same class.***

From the results obtained, it follows that the $L(\tau_n)$ family of languages shows strong immunity to closure relative to operations, traditional for ***TFG*** along with other operations that are interesting from standpoint of ***TFG*** itself and a number of interesting applications. In this regard, it is very interesting to compare ***L***–systems and τ_n–grammars with each other. So, the languages of the ***L***–family as a whole relative to $L(\tau_n)$–languages have complete immunity to traditional closure operations. An interesting enough discussion of the reason for the differences between ***L***–systems and τ_n–grammars can be found, for example, in [113,136-138,141,183].

The relationship was established [141] between the families of parallel formal languages $L(\tau_n)$ and $L(T_n)$, determined by means of parallel τ_n– and T_n–grammars of classical ***1–CA*** and non–deterministic ***1–CA*** models respectively in the *Chomsky* hierarchy. In order to better understand the place of languages $L(\tau_n)$ in this hierarchy, ***<k, p>****–Lindenmayer* languages were also included [121,141]. In [161], the visual diagram determines the relationship between the parallel languages $L(\tau_n)$ and $L(T_n)$, defined by the classical and non–deterministic models ***1–CA*** respectively, within the framework of the generally recognized hierarchy of the formal *Chomsky* languages; more precisely, the location of the formal languages $L(\tau_n)$ and $L(T_n)$ in the generally recognized hierarchy of *Chomsky* languages. Along with the above languages defined by ***CA***–models, the hierarchy includes well–known languages such as regular, context–free, context–dependent, ***<k, p>****–Lindenmayer* languages, recursive and recursively enumerated languages. Many interesting properties of $L(\tau_n)$– and $L(T_n)$–languages concerning various operations with them are presented in our works [113, 136-138,141] and in the references to primary sources contained in them.

Finding a certain class of recognizers or acceptors that allow languages generated by grammars is the traditional approach in ***TFG.*** Obviously, a good automatic model of a certain family of formal languages gives it a fairly strict characteristic. All reasonable models of this type have a finite automaton as a control device. Therefore, the family of formal languages allowed by similar models must be closed in relation to the intersection operation with regular sets of words. Of this standpoint, various classes of $L(\tau_n)$ languages were studied and the following result was obtained in this direction [10,11,113,136-138,141,183]:

Class of all parallel $L(\tau_n)$–languages is not closed with respect to the intersection operation with regular sets of finite words.

So, it follows from this result that it is impossible to find an automaton model of acceptors in the standard sense with respect to the class of the parallel $L(\tau_n)$–languages. A study was made of the $L(\tau_n)$–languages to preserve the property of being again the language of the same class when narrowing or expanding it with some finite subset ***S*** of words from the set $C(A, 1, \phi)$. Together with other results on the nonclosure of the class of parallel languages $L(\tau_n)$ regarding a number of important set–theoretic operations, the results [136-138] confirm a rather strong nonclosure of the class of $L(\tau_n)$–languages in this direction. This property significantly distinguishes the class of $L(\tau_n)$–languages from the traditional families of formal languages considered in the classical ***TFG.***

One possible way to research the structure of τ_n–grammars is to impose partial constraints directly on the definitions of their components and then study the influence of these constraints on the languages generated by grammars. A number of results in this direction is presented in [161]. So, formulaic sequences of words are examples of $L(\tau_n)$–languages, in which the words forming them in some respect contain a history of their development. A $L(\tau_n)$–language is a *formulaic* language if an appropriate τ_n-grammar generates a *formulaic* sequence of words *(configurations)*. It can be shown, the parallel $L(\tau_n)$–language generated by an appropriate τ_n–grammar determined by the *linear* classical model ***1–CA*** is *formulaic* language [113,141]. This is another kind of general characteristic of the generating capabilities of such class of ***CA***–models, which is generalized to the case of ***d***-dimension. There are quite a few very complex examples of formulaic languages defined by classical ***CA***–models. We studied a number of the formulaic languages. The concepts of *formulaic* grammars and languages introduced are of interest in studies of syntactic structure of parallel languages generated by τ_n–grammars. At that, these concepts are quite closely related to the use of classical ***d–CA**s* models ***(d ≥ 1)*** as a modelling environment for various processes, objects and phenomena. Therefore, there is a pressing problem of determining the formularity of a $L(\tau_n)$–language; this problem, in our opinion, is unsolvable. In addition, despite a number of the results obtained, there is currently a rather scant information regarding the *formulaic* representation of $L(\tau_n)$–languages, so studies in this direction are very desirable from a structural standpoint.

By introducing parallel τ_n–grammars, it is quite natural to compare their generating capabilities with previously studied formal grammars of other types and classes. The results obtained in this direction make it possible not only to obtain from many standpoints quite interesting comparative estimates of a new class of parallel grammars determined by classical ***CA***-models, but also, on the other hand, to evaluate parallel τ_n–grammars and the formal parallel languages generated by them. For example, *E.S. Shcherbakov* [164], dealing with the issue of a mathematical modelling apparatus for developmental biology at the cellular level, introduced a new class of parallel grammars, which was later called ***Sb(m)***–grammars. It has been shown that according to the generating capabilities, ***Sb(m)***–grammars and τ_n-grammars are equivalent [138]. Discussion of a number of issues concerning the relationship of the τ_n–grammars with some other types of parallel grammars (such as *isotonic* structural grammars, parallel *spatial* grammars, parallel programmable spatial grammars, etc.) together with a number of traditional grammars can be found in [155]. Since the class of languages $L(\tau_n)$ is the own subclass of the class of languages of *A. Lindenmayer*, that are generated by ***L***–systems, a number of questions arise which relate to a more detailed identification of relations between both classes of these formal languages. In particular, we have shown that ***any Lindenmayer L–system is modeled by the corresponding classical model 1–CA, but, generally speaking, not in real time, and vice versa.***

Along with parallel grammars defined by classical ***CA**s*, we investigated a lot of issues related to parallel grammars defined by *non–deterministic* cellular automata. Some of these can be found in [161]. The solvability problems play a very important role in the modern mathematics. In this regard, we considered the question regarding the solvability problems for parallel grammars determined by classical cellular automata. The main results in this direction can be found in [113,161,182]. Note, meanwhile, that we did not pay such close attention to the issues of τ_n–grammars.

5.6. Modeling problem in classical cellular automata and certain related issues

Simulation in classical ***d–CA****s* models ***(d ≥ 1)*** is of great theoretical and applied interest. A significant number of works containing rather many interesting results are devoted to this problem. One of the areas of study in this field is related to the modeling of one ***d–CA (d ≥ 1)*** by the other: real-time modeling; modeling with suppression of certain properties of the modelled ***d–CA,*** simplification of parameters of the simulated model, etc. If in the previously discussed areas of ***CA***–problematics there was practically no optimization problem, then in the simulation it is already assumed to use a certain optimization. Many researchers have been quite actively involved in modeling in classical ***d–CA****s* [3,10,11,154,164]. This issue has also been considered by us in various aspects.

First of all, we note *two* modelling methods in ***CA****s*. Like the founders of ***CA***–problematics *(John von Neumann, S. Ulam, A. Burks, J. Holland, E. Codd, E. Banks, H. Yamada, etc.),* a fairly large number of researchers in this direction applied ***CA***–models directly for theoretical and especially applied modelling tasks, providing them with the required functioning rules with embeddind in them of simulated algorithms and objects. This approach is clearly constructive when, in a desired ***CA***–environment, a single simulated task can be reduced to the composition of the subtasks that make up it. One typical modeling method of this type is to create a number of blocks of unit automata in the ***CA***–environment that perform certain functions and interact with each other by exchanging control impulses through organized information channels formed by elementary automata of the ***CA***–environment. This approach determines the direct embedding of the simulated tasks in ***CA****s* and in many cases effective.

Whereas the *second* approach uses ***CA***–models as certain formal parallel information processing systems, presenting a more general level of the modeling of the investigated algorithms. In this regard, both modelling approaches based on the ***CA***–method can be compared to a certain extent with well–known modeling approaches based on *Turing* machines and *Markov* algorithms or some other formal algebraic symbolic processing systems in finite alphabets. If the *first* approach is most suitable for the research of applied aspects of modeling based on ***d–CA****s* ***(d ≥ 1)****,* then the *second* approach forms the basis of a formal research of the constructive and computational capabilities of ***CA***–models as certain abstract systems for parallel processing of information, which at the axiomatic level will provide the properties of homogeneity and locality, while at the program level – the *reversibility* property of ***CA***–dynamics. Both methods can be complementary with a reasonable degree of admissibility. Note, it was the *second* approach that mainly focused our attention in this direction.

By introducing the concept of modeling one ***CA***–model by another of the same dimension, a number of results were obtained, in particular, a result useful for modeling tasks in classical ***d–CAs*** $(d \geq 1)$ [113,148,183]:

A classic d–CA model (d≥1) with neighborhood template in the form of hyperparallelepiped of size $n_1 x n_2 x \ldots x n_d$ ***and alphabet*** $A = \{0, 1, ..., a-1\}$ ***is simulated in real time 1/n by a model of the same d–dimension with alphabet A* and neighborhood template in the form of hypercube with edge of length two under the following conditions:***

$$n = \max_{k=1..d} \{n_k\} - 1 \qquad \# A^* = \sum_{k=1}^{n} a^{k^d}$$

A characteristic property of a simulating ***CA***–model is its inheritance of a number of basic dynamic properties of the simulated model. The given circumstance is quite significant, allowing, first of all, at the theoretical level, to investigate the dynamics of classical ***d–CAs*** models $(d \geq 1)$ with simple neighbourhood indices, with extension of the previously obtained results to more general types of classical ***CA***–models. It is shown, the results obtained are characterized by the fact that classical models with large neighbourhood template and small states alphabet can be simulated by models of the same type with smaller neighbourhood template along with large states alphabet, and vice versa.

Since the classical ***d–CAs*** models $(d \geq 1)$ are parallel words processing algorithms of ***d***–dimension in finite alphabets, it is quite interesting to compare them with the well–known formal sequential algorithms. One approach of this type is to model an algorithm of one type by another, and vice versa. In [148,161], we presented the concept of ***T***–modelling and on its basis , a number of questions of simulation by classical models ***1–CAs*** of such well–known computational algorithms as *TAG*–systems, *LAG*–systems, regular *Büchi* systems, *SS*–machines, the normal *Markov* algorithms, *Post* production systems, etc. were discussed in sufficient detail, and vice versa. In the results, along with the application of the ***T***–modeling principle, an optimizing technique was also used, which made it possible to obtain rather optimal relations between the main parameters of the modelling and modelled algorithms. In particular, an interesting enough consequence arises from simulation of an arbitrary *SS*–machine by a suitable ***1–CA*** model, namely [10,11,113,155,183]:

There are classical 1–CA models whose sets of the finite configurations degenerated into zero configuration are creative.

So, there are classical ***CA***–models whose sets of the finite configurations generated into zero configuration are *non–recursive*. In this connection, an interesting question arises about the existence of classical ***CA***–models whose similar sets of finite configurations are *simple* or *maximum*, and what are the values of the basic parameters for ***1–CA*** models of this type. At the same time, on the basis of the mentioned simulation a number of results were obtained on unsolvability of some questions of dynamics of classical ***CA***–models [10,11,113,147,161,182-196].

From an applied standpoint, modelling of classical ***d–CA*** models ***(d ≥ 1)*** by binary models of the same dimension is of particular interest, firstly in computer sciences and in a number of other interesting applications. The optimization problems in all areas are quite complex. The above problem is not an exception; therefore another method of investigation was used to solve it [113,148]. The proposed approach yielded the following result having a number of both theoretical and applied applications:

A classical model d–CA (d ≥ 2) is 1–simulated by the appropriate binary CA–model of the same dimension and neighborhood template of size L:

$$L = (L_1)^{d-1}(L_d+1)\prod_{k-1}^{d}(p_k+1); \quad L_1 = \left[V = \sqrt[d]{log_2(a-1)+2}\right]; \quad L_d = L_1 + \left[2(V-L_1)\right]$$

where A = {0,1,...,a–1} – states alphabet and $p_1*p_2*p_3*...*p_d$ is the size of minimum hyperparallelepiped, that contains neighborhood template of the simulated CA–model, provided the condition $log_2 log_2 4(a-1) \geq d$.

Therefore, the edge of the ***d***–dimensional neighborhood template of the simulating ***d–CAs (d ≥ 2)*** under this condition is asymptotically reduced by $\sqrt[d]{log_2(a-1)+2}$ time with the increase of ***d***–dimension. Previously, the non-equivalence of models ***1-CAs*** and ***d-CAs (d≥2)*** with respect to some phenomena was noted; this also applies to the modeling problem in the classical ***d–CA***–models ***(d ≥ 1)***. We paid special attention to this point, so the modeling method began to proceed from the influence of dimension of the classical model ***d–CA (d ≥ 1)*** on the optimization factor. Therefore, more optimal modeling required slightly different approaches.

So, for the ***1***–dimensional case, an optimal technique was proposed that takes into account the specifics of the functioning classical models as much as possible ***1–CAs***. Such technique is based on the principle of the maximum approximation of the characteristics of simulating models to the main characteristics of potentially optimal simulating models. At the same time, simulating models whose base parameter values may not be achievable, but which can serve as a rather good reference for promising researches in this direction and for evaluating the parameter values of the previously created simulating models, are considered potentially optimal models [172]. In particular, a ***CA*** model with neighborhood template of size $L_{opt}=(n+1)[log_2 a]+2$, that, meanwhile, is unattainable, is considered as potentially optimal binary simulation model for classical models ***1-CA*** with states alphabet $A=\{0,1,...,a-1\}$ and neighborhood template of size n. From standpoint of this assessment, there is an obtaining problem of the optimal classical model ***1–CA,*** which is closest to the potentially optimal model of the same dimension. We defined simulating binary model ***1-CA*** with neighborhood template of size $L = (n+1)[log_2\, a + 1 + \omega] + 2$, where $0 < \omega < 1$ [173]. From the given assessment it is easy to obtain conclusion on quite satisfactory similarity of the received simulating model to some standard model even at the moderate cardinality of states alphabet and size of neighbourhood template of the simulated classical model ***1–CA.*** Studies in this direction made it possible to formulate the result:

A classical d–CA model with states alphabet $A = \{0,1,2,3, ..., a–1\}$ and neighbourhood template which is in the minimum hyper parallelepiped $p_1*p_2*...*p_d$ of d–dimension is 1–simulated by the appropriate binary classical d–CA model ($d \geq 1$) with neighbourhood template of the size $L = (p_1+1)[log_2\, a + p_1 + \lambda]*p_2*p_3*p_4*...*p_d$ where $\lambda = 4$ for values $a \leq 2^{19}$ and $\lambda = 5$, otherwise.

The proof method allows to simulate classical models ***1–CAs*** which have a large enough states alphabet and small neighborhood templates quite efficiently, using binary classical models ***1–CAs*** with acceptable sizes of neighborhood templates.

In connection with definition of *universal computability* based on the ***T***–modeling concept, a rather important question arises about the minimum complexity of classical ***CA***–model, that ***T***–simulates the universal *Turing* machine, or in more general statement about the simplest classical ***1–CA*** model, that has universal computability. As a measure of the complexity of the universal ***d–CA*** model ($d \geq 1$), it is quite natural to use the $d*a*n$ indicator, where ***3*** parameters determine the values of basic parameters of such model: *dimension (**d**), cardinality of states alphabet (**a**)* and *size (**n**) of neighborhood template*. For classical ***1–CA*** models, the best result in this direction was obtained by *A.R. Smith*, who proved the presence of universal models with such values for $a*n$ indicator as: ***2*40, 3*18, 6*7, 8*5, 9*4, 12*3, 14*2.*** The best result of similar type for universal classical models ***2–CA*** was obtained by *E. Banks*, who proved the existence of the universal models with the value $d*a*n = 2*2*5 = 20$ using the infinite initial configuration of the simulating model whereas $d*a*n = 2*3*5 = 30$, otherwise. In turn, *A. Podkolzin* proved the existence of universal models ***2–CAs*** with values such as $d*a*n = 2*2*9 = 36$ and $d*a*n = 2*3*5 = 30$. We showed [147], the universal classical model ***1–CA*** with value $a*n = 14*2$ is ***1***–simulated by a binary model ***1–CA*** with neighborhood pattern of size $2[2log_2\, 16 + 1] – 2 = 16$, i.e. for a simulating model ***1–CA*** the value $a*n = 2*16$ is permissible, resulting in the following result:

There are universal classical 1–CAs models with value indicator a*n = {2*16 | 3*11 | 4*8}; such classic models were obtained by us as a result of simulation in strictly real time.

Note that in this direction we have obtained a number of other interesting results [161]. At the same time, it is interesting to study the properties of such universal models ***1–CA****s* from standpoint of the nonconstructability problem. In this direction, we got a rather interesting result [113,148]:

There are universal classical 1-CA models that have nonconstructability of all four types: NCF, NCF–1, NCF–2 and NCF–3.

A number of aspects of the discussion of modelling techniques, together with related issues, can be found in [113,141-148]. The modeling issue of classical ***d–CA****s* models ***(d ≥ 1)*** by ***CA***–models of the same class, but with decrease in the dimension of the analogue model, is of significant both theoretical and applied interest. [174] presented and analyzed an interesting approach to the modeling problem of classical models ***3–CA****s* by models ***2–CA.*** A generalization of the approach allows to model ***d-CA (d ≥ 3)*** by models ***2–CA.*** Meanwhile, this approach does not work for the ***1***-dimensional case and does not allow to model an arbitrary ***2-CA*** model by the corresponding model ***1–CA.*** Our approach enables the simulation of classical models ***2–CA****s* by models ***1–CA****s* of the same type.

The dynamics of finite configurations in classical 2–CAs are simulated by the corresponding classical models 1–CA with the Neumann–Moore neighborhood index. A classical 2–CA with the simplest neighborhood index and cardinality states alphabet G is simulated by the appropriate classical 1–CA with the Moore neighborhood index and the cardinality states alphabet $3G^2 + 5G + 27$.

Moreover, our study allows to formulate a rather interesting statement:

Dynamics of finite configurations in classical model d–CA (d ≥ 1) with an arbitrary states alphabet are simulated by the corresponding binary classical 1–CA or non-binary classical 1–CA with the Neumann-Moore neighborhood index.

Parallel algorithms defined by the classical ***d–CA****s* models ***(d ≥ 1)*** play a rather significant role in the formal description of a number of biological development processes and various programmable systems that are based on computational homogeneous structures. Due to the undeniable interest in solving important problems of designing multiprocessor languages, the study of formal language models, which operate strongly in parallel, is of particular importance. We determined [155] a wide class of parallel algorithms ***1–PACA*** and studied issues of their complexity regarding a number of known formal words processing algorithms *(configurations)*.

Since in the theory of formal algorithms a lot of attention is paid to the complexity of calculations, a result was obtained regarding the class of parallel algorithms ***1–PACA*** in this direction [10,11,113,155,183]:

A partially recursive dictionary function F defined in a finite alphabet A, is PACA–computabled in the extended alphabet A* = {***b***} ∪ ***A (b∈A).***

We analyzed the parallelism issues of the ***1-PACA*** algorithm class and, as a result of the analysis, identified the most interesting areas for further study: *parallelism classes, refinement of internal essence of parallelism, selection of algorithms most suitable for effective implementations in the computing* ***CA****–models, etc.;* as part of this, the solution of a number of mass problems in the class ***1–PACA*** was investigated [113,141-148]. In particular, it is shown that ***if a two–way stack automaton allows a set of finite words in w steps, then a suitable parallel algorithm 1–PACA can allow the same set of words in no more than*** $2w^2$ ***steps.*** So, in terms of the parallel algorithms ***1–PACA*** we get a essentially better result on time complexity relative to the known standard result. In general, parallel ***1–PACA*** algorithms for a lot of computational algorithms have been shown to produce much better time results than on *Turing* machines. Meantime, compared to the theory of sequential algorithms, the theory of parallel calculations supported by ***CA****s* is not developed in such detail.

It is known that simulation in the classical ***CA****s* is a multifaceted problem that includes such rather complex issues as real–time modeling, optimal modeling according to selected optimization criteria, methods to simplify the modeling process, obtaining estimates of the complexity of mutual simulation of ***CA****s*, modeling individual objects, processes, phenomena and algorithms, simulating in certain classes of ***CA****s*, etc. The simulation issues in classical ***CA****s* without any additional conditions for simulating ***CA****s* were above discussed. Now let`s take a brief look at some simulation questions when simulating classical ***CA****s* are subject to certain constraints that have one meaning or another. The issue is significant enough.

The study of the dynamic properties of classical ***CA****s* in connection with the type of their local transition functions ***(LTF)*** is of undeniable interest. We identified two large classes of ***CA****s* both with *symmetrical* ***(SF)*** and *asymmetric* ***(ASF)*** *LTFs* and considered a number of questions regarding their important properties. In particular, it has been shown the significant differences exist with respect to constructive capabilities and sets of the nonconstructible configurations in classical ***CA****s* with *asymmetric* and *symmetric LTFs* that undoubtedly needs to be considered in many model applications [155]. Quite a few processes have a pronounced asymmetric character *(although, meanwhile, at their base at the lowest levels there may be elements of different symmetry levels)* and they can be relatively simply embedded in the classical ***CA**s* with asymmetric *LTFs*. Whereas embedding them into classical ***CA**s* with symmetrical *LTFs* requires, at times, a significant complication of the basic parameters of the second ones *(alphabet cardinality, simulation time, neighborhood pattern size)*. In this direction, along with certain our interesting results were obtained results by *H. Schwerinski* and *Yu. Kobushi* [113,154,161,164,182-196].

We would like to once again note that both classes ***SF*** and ***ASF*** of the classical ***CA**s* have many specific features, however, based on practical considerations, there are two main differences between them: ***CA**s with symmetric LTFs appear to us much easier to implement and particularly are of interest in terms of different biological interpretations, whereas **CA**s with asymmetric LTFs are generally substantially better adapted to simulate different processes and algorithms, i.e. have a greater degree of constructive capability for a number of key indicators*. As our experience shows, for pronounced asymmetric processes, overall, it is impossible to quite satisfactorily solve many optimization problems in symmetric ***CA**s*. When simulating in classical ***d-CA**s (d≥1)*, the optimal modeling problem of various objects, algorithms or phenomena is quite important. At that, optimization is usually considered with respect to such basic parameters of the modeling ***d–CA*** as neighborhood pattern size, alphabet cardinality, dimension and simulation time. In this regard, we have obtained a lot of rather interesting results, including *reversibility* issues [141-148,161].

A rather important direction in ***CA***–problematics is the study of modeling issues of classical ***CA**s* under certain conditions, e.g., in the absence of one or another of nonconstructability type in the modeling ***CA**s*. So, one of our approaches allows to simulate classical ***d–CA**s (d ≥ 1)*, including ***CA**s* having nonconstructability of ***NCF*** type, by classical ***(d + 1)–CA**s* that do not have nonconstructability of ***NCF*** type, allowing to formulate an interesting result for a number of applications and theoretical studies.

An arbitrary classical d–CA (d ≥ 1) with states alphabet A is 1–modelled by classical (d + 1)–CA with the same alphabet A; at that, appropriate simulating (d + 1)–CA does not have nonconstructability of NCF type and preserves history of dynamics of an arbitrary finite configuration of the simulated classical d–CA.

Without disturbing commonalities with the essence of such approach that is based on classical ***1–CA**s* can be familiarized, for example, in [161]. At the same time, with a detailed description of the simulation algorithm itself underlying the proof of this result can be familiarized in [142]. It should be noted that the above result along with the *T. Toffoli* result [164] will determine high enough price for such simulation – an increase in the dimension of the modeling ***CA*** relative to the dimension of the modelled classical ***CA.*** It follows from our results [161] that classical ***d–CA (d ≥ 1)*** within the dynamics of structure–periodic and/or finite configurations is simulated by a suitable classical ***1–CAs*** with the simplest neighborhood index. In addition, the approach taken in the proof provides simulation of dynamics only in classical ***d–CAs (d ≥ 1)*** and on the more general case of ***CAs*** modeling does not apply. In view of this the following proposal may be formulated alongside the above result:

A classical d–CA (d ≥ 1) within the dynamics of finite and/or structure–periodic configurations is simulated by the appropriate classical 2–CA, that does not possess the nonconstructability of NCF type and has the simplest neighborhood index X = {***(0,0), (0,1), (1,1)***}.

Note that by *reversibility*, a number of researchers mean the absence of mutual erasability for classical ***CA*** *(nonconstructability of the NCF type)*, while we also mean and the absence of nonconstructability of the *NCF–1* type by reversible ***CAs****;* justification of such prerequisite is presented in [142,161]. Meanwhile, the approach used by *T. Toffoli* not only requires increasing in the dimension of the simulating ***CA,*** but also don`t relieve it from nonconstructability of ***NCF–1*** type, not allowing to fully take into account the dynamics of such simulating ***CAs*** that are reversible to the full extent. Moreover, *T. Toffoli* used some structural approach to create reversible ***CAs****,* representing the elementary ***CA*** automaton by a simple logical scheme of three elements. Meanwhile, analysis of this approach shows, the reversibility property is achieved due to an implicit increase in the states alphabet cardinality and refers to some subset of it [164]. In the meantime, *real* reversibility regarding the expanded alphabet has not been achieved. On the other hand, by real *reversibility*, we will mean the dynamics *reversibility* of classical ***CAs*** relative to the set $C(A, d, \phi)$***.*** Here it is appropriate to briefly discuss two levels of *reversibility*, namely, *real* and *formal*. ***Formal*** level refers to the reversibility of finite configuration c^*, namely the existence for configuration $c^* \in C(A, d, \phi)$ of such single configuration $c' \in C(A, d, \phi)$ irrespective of the set $C(A, d, \infty)$ that there is a relation $c'\tau^{(n)} = c^*$. While ***real*** level refers to reversibility with respect to finite configurations; that is, the existence for a finite configuration c such single finite configuration c^* of only of the set $C(A, d, \phi)$, that the relation $c^*\tau^{(n)} = c$ takes place. Thus, depending on the existence criterion of ***NCF*** based on the concept ***MEC*** or ***γ–CF,*** it is easy to make sure that the presence of *formal* reversibility can entail *real* irreversibility whereas the opposite, generally speaking, is wrong.

Obviously, *real* reversibility in classical ***CAs*** entails *formal* reversibility, while the inverse statement, generally speaking, is incorrect. One of the motivations for introducing the concept of the *real* reversibility of the dynamics of finite configuration in classical ***CAs*** is natural requirement – the predecessor in the prehistory $\{c\tau^{(n)k} \mid k = -1, -2, ...\}$ of a configuration $c \in C(A,d,\phi)$ should be calculated in a finite number of steps. In particular, in the assumption of belonging of completely zero configuration c_o = `□` to the set $C(A,d,\phi)$, the existence in a classical ***CA*** of nonconstructability of *NCF–1* type makes it *really* irreversible. From standpoint of two types of mutually erasable configurations, one can present the criterion of two types of *reversibility* in classical ***CAs*** [113,141-148,161,182-196]:

A classical d–CA (d ≥ 1) is formally (really) reversible if and only if it don`t have MEC (MEC–1) pairs; i.e. it don`t have nonconstructability of NCF (NCF and NCF–1) type.

In connection with this, a rather interesting question arises: ***Is it possible to simulate an arbitrary classical d–CA (d ≥ 1) using reversible d–CA?*** In turn, this question raises a number of related issues that to some extent describe the *reversibility* problem in the classical ***CAs***. In general, similar questions constitute the general problem of simulating arbitrary classical ***d–CAs (d ≥ 1)*** using classical ***CAs*** of the same dimension, suppressing some properties of simulated ***CAs***. In addition, in relation to the *formal reversibility* characterized by the existence of nonconstructability of the ***NCF–1*** type in ***CAs***, a result is obtained that plays a well–defined role in studies of the dynamic properties of classical ***d–CAs (d ≥ 1)*** [141-148]:

A classical d–CA (d ≥ 1) is 1–simulated by a suitable d–CA of the same type with minimal expansion of the states alphabet; at the same time, the simulating d–CA does not have nonconstructability of the NCF–1 type in the presence of nonconstructability of the NCF–2 type.

A detailed description of the modeling algorithm underlying the proof of this result can be found, for example, in [113,142]. This is much more difficult in the case of nonconstructability of the ***NCF*** type, that together with ***NCF–1*** type forms the basis of the reversibility concept in classical ***CAs***. As part of the study of this question, the concept of ***WM***–modeling was defined, covering a rather wide class of methods of simulating one classical ***CA*** with another of the same class and dimension. On this basis, a result is obtained that to a certain extent characterizes the possibilities of modeling problems and is useful in a number of theoretical researches.

A classical d–CA (d ≥ 1) cannot be WM–modeled by the corresponding reversible CA (in terms of nonconstructability of the NCF type) of the same class and dimensionality.

In the process of research, we have defined the concept of ***W***–modeling, which significantly extends the concept of ***WM***–modeling and covers a fairly wide class of known and potentially permissible modeling methods in classical ***CA**s*. However, and this did not make it possible to positively solve the problem of *modeling* by the appropriate classical reversible ***CA**s* of the same dimension, as evidenced by the following main result [113]:

A classical d–CA (d ≥ 1) cannot be W–modeled by reversible d–CA (in the sense of absence for it the nonconstructability of the NCF type) of the same class and dimensionality.

Meanwhile, using the results of *K. Morita*, *J. Dubacq* and a number of others [164] together with ours, it is possible to prove a rather interesting result [10,11,113,161,182-196]:

An arbitrary classical d–CA (d ≥ 1) can be simulated by an appropriate formally reversible classical 1–CA.

Our other results on the multi–aspect modeling problem in ***CA**s* including mutual modeling with suppression of certain properties of simulated ***CA**s* can be found in our publications [113,141-148,155,161,182-196].

The *reliability* problem of ***CA**s* of this type, consisting of real elementary automata, relates to some extent to the general problem of simulation in classical ***CA**s*. Meanwhile, it has so far been assumed that ***d–CA**s **(d ≥ 1)*** are a purely abstract model, whereas in real conditions the work of ***CA***–models can undergo various kinds of disorders, which can lead to rather undesirable consequences. This poses a rather important problem for the ***CA*** organization, which would, in many important cases, correct possible failures that occur during the operation of real ***CA***–models. We will call a ***CA***–model *self–correcting* if the model during operation has the ability to eliminate the consequences of failures in the operation of elementary automata and their connecting information channels. We have proposed some methods for organizing the functioning of real ***CA***–models based on self–correcting computing structures [10,11,113,141,183].

Further research into the *self–healing* problem of the real ***CA***–models of different types is of considerable applied and cognitive interest, and this direction should be given appropriate attention. The study of the stability of real ***CA***–models to failures of various kinds can quite be attributed to this direction. At the same time, the proposed approaches are not only formal in nature, but also allow us to consider the *reliability* problem of cellular systems of various nature from formal standpoints [141]. The proposed techniques for correcting real ***CA***–models are of particular, first of all, theoretical interest, bearing the features of a certain common basic approach, while for practical application they are perhaps not effective enough taking into account the use of the necessary resources. Therefore, to solve real practical problems, it is necessary to develop more effective methods for correcting failures [10,11,113,148,161,182-196].

5.7. The decomposition problem of global transition functions in classical cellular automata *(CA–models)*

The decomposition problem of *global transition functions (**GTFs**)* for ***CA***–models is of considerable theoretical and applied interest. The goal of a ***GTF*** decomposition is to identify effective procedures which allow, based on a predetermined ***GTF,*** to determine the composition of simpler functions whose composition is equivalent to the initial ***GTF.*** The given problem is also directly related to issues of constructive complexity, that play a rather important role in specific implementations of ***CA***–models of various kinds. The first setting and the first results on the decomposition problem go back to *S. Amoroso* and *J. Epstein* [164], who proved that in the set of all binary ***1***–dimensional ***GTFs*** there are functions that are not represented as a composition of finite number of the simpler functions of the same type and class. Then, *J. Buttler*, using a rather simple numerical procedure, showed that in the set of all ***GTFs*** of ***d***-dimension ***(d≥1)*** there are also functions which are not represented as the so–called minimum compositions from a finite number of the simpler ***GTFs*** from the same set of functions [164]. In certain our works, the decomposition problem has been further developed [113,141,155-161]; the results obtained in this direction made it possible to consider this problem from new rather interesting standpoints. First of all, the *decomposition problem* relates to some extent to the *complexity problem* of global transition functions:

Can an arbitrary global transition function $\tau^{(m)}$ be represented by some composition of a finite number of the simpler GTFs of the same class and the same alphabet?

At that, we will say that a global transition function $\tau^{(n)}$ is simpler than a global function $\tau^{(m)}$ *(both global functions are defined in the same finite alphabet and the same dimension)* if $n < m$; $n < m$ defines the relationship between the number of automata maked up the neighborhood templates of both ***CA***–models. It turned out that problems such as complexity of the finite configurations, completeness problem for polygenic ***CAs*** and decomposition problem of global transition functions are rather closely related, presenting a promising and extensive field for further researches. It is easy to verify that in the general case, an arbitrary global transition function $\tau^{(n)}$ cannot be represented by a composition of finite number of the simpler global transition functions of the same class and in the same finite alphabet. Our first results regarding the decomposition problem are based on earlier results on nonconstructability problem in classical ***CAs*** and solved the problem for classical ***1–CAs*** [145]. The main result in this direction was to prove the existence of ***1***–dimension ***GTFs*** with arbitrary neighborhood indices and finite states alphabets for which decomposition problem has the negative solution.

In general, the *decomposition problem* of global transition functions $\tau^{(n)}$ ***(d–DPF; d ≥ 1)*** is defined by the possibility of representing an arbitrary global transition function in the form of a composition of finite number of the simpler functions of the same class and in the same alphabet $A = \{0,1,2,3, ..., a-1\}$, namely:

$$\tau^{(n)} = \tau^{(n_1)}\tau^{(n_2)}\tau^{(n_3)}...\tau^{(n_p)} \quad (n > d+1;\ n_j < n;\ j = 1..p)$$

wherein the global transition functions $\tau^{(n)}, \tau^{(n_j)}$ $(j = 1..p)$ have the same dimension and are defined in the same alphabet; in addition, the multiple occurrences in the above representation are also quite allowed for global transition functions $\tau^{(n_j)}$. For the case of ***1***–dimensional global transition functions $\tau^{(n)}$ the relations $n = \sum_j n_j - p + 1$ and $(\forall j)(n_j \in \{2,3,4, \dots, n-1\})$; $j = 1..p$ exist for the above representation. It is shown [113,146,177] that an arbitrary global transition function with respect to representability in the above form satisfies one of three opportunities: *(**1**) it don`t has the above representation, (**2**) has a single representation and (**3**) has more than one representation*. In particular, if a global transition function has a single representation, then all global transition functions which make up such representation will have negative ***d-DPF*** solution. This is one of the easiest approaches to proving the existence of negative ***d–DPF*** solutions. Taking into account the finite number of global transition functions in the representation and a finite number of these functions in the case of finite alphabet of states, it is easy to show, for example, based on the search method *(albeit very bulky)*, that it is possible to solve the decomposition problem of an arbitrary global transition function on simpler functions, including the case of uniqueness of the representation of ***GTFs***.

Meanwhile, it is likely that with the advent of quantum supercomputers, it will be necessary to reconsider the temporal complexity of a number of problems *(by transferring them from a class of unsolvable to a class of solvable or difficult to solve)*, currently unsolvable due to the required time resources; the same applies to a number of applied problems from theory of classical cellular automata – the main purpose of our studies.

Along with ***d-DPF,*** a *special* representation of global transition functions in the form of a composition of finite number of simpler functions is of particular interest. By special we will mean any representation of a global transition function $\tau^{(n)}$, provided that the original function and functions $\tau^{(n_j)}$ which make up its decomposition are selected from a given class of functions with imposition on them of some special restrictions that have a certain interpretation. In particular, we are naturally interested in the relationship between the nonconstructability properties of an arbitrary global transition function and global transition functions that compaund its representation. In this direction, we obtained a number of results [113, 141,146,157,179,183], in particular, shown:

A global transition function $\tau^{(n)}$ set in the finite alphabet has the NCF nonconstructability if and only if at least one global transition function $\tau^{(n_j)}$ in its representation will have NCF; if at least one global transition function $\tau^{(n_j)}$ has NCF–1, their composition $\tau^{(n)}$ will have NCF or/and NCF–1 nonconstructability. There are global transition functions other than linear ones which have universal or substantial reproducibility in the Moore sense of finite configurations; the same property will kept when their composition is with linear GTFs. Such compositions with linear GTFs may include GTFs that do not have nonconstructability of NCF type in the presence of the NCF–1 nonconstructability.

A lot of results in this direction can be found, for example, in [113,161]. It also presents a number of our results on the decomposition of special global transition functions in classical ***CA****s*. A one–dimensional case was considered in sufficient detail, including a number of interesting special classes of one-dimensional ***CA****s*. Of the ***d***–dimensional ***CA****s* ***(d ≥ 2)***, in particular, a class of ***CA***–models with refractority, having bio–medical interpretations and used to research the problems of recognizing images, excitable media, properties and topology of digital figures, etc. was rather detally investigated. The decomposition problem in the class of all ***CA****s* ***(d ≥ 2)*** with refractority has been shown to have the negative solution. It suggests a negative ***d–DPF*** solution in the class of all global transition functions with refractory, regardless of excitation threshold and refractory depth. Other rather interesting special representations of global transition functions by the composition of a finite number of simpler functions are considered in our works [10,11,113,141,142,157,179,183].

Above we have discussed certain questions of decomposition problem relating to a number of special classes of global transition functions, now we will consider some approaches to solving the general decomposition problem which are based on the application of the results and methods of the theory of functions of algebra logic, ***a***-valued logics, formal apparatus of group theory, algebras and semigroups, as well as on a rather essential generalization of the method of solving ***d–DPF (d ≥ 1)*** based on the study of the nonconstructability problem in classical ***CA****s*. Along the way, the decomposition problem of ***GTF****s* is found to be rather closely related to the complexity problem of finite configurations in classical ***CA****s* and to the completeness problem for polygenic ***CA****s*. Approaches and methods for ***d–DPF*** solution are of interest in the study of some other ***CA****s* issues and a number of their important enough applied aspects.

First of all, to solve the decomposition problem we proposed an approach [179], based on the use of the *C. Shannon* function, that was introduced to assess the complexity of implementing the functions of algebra of logic. On the other hand, in the general case of states alphabet of a classical ***CA,*** it is impossible to directly generalize the above results regarding binary ***GTFs***, so we are forced to first turn to ***a***–valued logic $(a > 2)$. Based on our approach used ***a***–valued logic, similarly to the binary case, negative solution to the decomposition problem was obtained in the case of global transition functions as a whole [113,148,183]. In addition, it is shown:

Among classical d–CAs (d ≥ 1) defined in the states finite alphabet and with arbitrary neighborhood indexes, there is an infinite set of global transition functions for which d–DPF has negative solution.

A similar result is rather easy to obtain based on our results above on the complexity problem of finite configurations in classical ***d–CAs*** $(d \geq 1)$. The impossibility of a positive ***d–PDF*** solution for an arbitrary global transition function allows to naturally introduce the complexity concept for the global transition functions themselves, similar to the case of finite configurations in classical ***d–CAs*** $(d \geq 1)$. Of analysis of the complexity concepts of finite configurations and ***GTFs*** in classical ***CAs***, it follows that this is based on the inability to exist in them some finite base sets for finite configurations and ***GTFs***, respectively. Moreover, in studies of the decomposition problem, some algebraic methods can also be used [141].

It is known that global transition functions that implement mapping the configurations of the set $C(A, d)$ to itself form some semigroup relative to the composition operation; let $L(a, d)$ denote the *semigroup* of all such global mappings of ***d***–dimension $\tau^{(n)}: C(A, d) \to C(A, d)$. It can be shown that $L(a, d)$ is a non–commutative semigroup with a group identity *(for complete certainty, defined by the relation* $\tau^{(2)}(x, y) = x$*)*, that leaves any global transition function $\tau^{(n)}$ unchanged within the leading variables. So, the study of the composition properties of global transition functions can often be reduced to the study of the appropriate properties of a suitable semigroup $L(a, d)$. It is shown [179] that the semigroup $L(a, d)$ contains one maximum group ***G,*** where maximum group is such group which is contained in the semigroup $L(a, d)$, not expandable by supplementing ***G*** with new elements from the set $L(a, d) \backslash G$. Considering the set $G(d)$ of all ***d***–dimensional ***GTFs*** whose respective global mappings $\tau^{(n)}: C(A, d) \to C(A, d)$ are one–to–one, it can be shown that the set $G(d)$ forms a group relative to the composition operation. The result allows us to apply group methods of studying the dynamics of classical ***d–CAs*** $(d \geq 1)$, that is, to reduce the study of a number of properties of such classical models to the study of the appropriate properties of group ***G(d),*** as well as subgroups that make up it. It is shown that the semigroup of mappings ***L(a, d)*** can be decomposed into combination of the disjoint semigroup ***L*(a, d)*** and the maximum group ***G(d),*** i.e., there are the following defining relations:

$$L(a, d) = L^*(a, d) \cup G(d) \quad \& \quad L^*(a, d) \cap G(d) = E$$

where ***E*** **–** a unit group consisting of only one unit element **–** *semigroup identity*. The solution of many issues concerning the possibilities of the semigroup ***L(a, d)*** can be reduced to solving the relevant issues for group ***G(d)*** or semigroup ***L*(a, d),*** which we did. Of our study on this way, it is easy to obtain an important consequence: ***If the group G(d) contains an infinite number of closed subgroups, then it cannot have a finite basis.*** The result can be transferred to the case of arbitrary algebraic semigroups. It is shown [10,11,113,142,157,161] that set of parallel global mappings $\tau^{(n)}: C(A, d) \to C(A, d)$ can be represented as a combination of ***7*** *disjoint* subsets having the following basic defining properties, which relate to a component of parallel global mappings **–** $\tau^{(n)}$:

G_1: GTFs $\tau^{(n)}$ possess the nonconstructability of 4 types NCF, NCF–1, NCF–2 and NCF–3 at the same time;

G_2: GTFs $\tau^{(n)}$ possess the nonconstructability of the NCF (NCF–3) and NCF–1 types without nonconstructability of the NCF–2 type;

G_3: GTFs $\tau^{(n)}$ possess the nonconstructability of the NCF (NCF–3) and NCF–2 types without nonconstructability of the NCF–1 type;

G_4: GTFs $\tau^{(n)}$ possess only nonconstructability of the NCF–2 type; at that, the global mappings defined by such GTFs are not one–to–one;

G_5: GTFs $\tau^{(n)}$ possess only nonconstructability of the NCF–1 type;

G_6: GTFs $\tau^{(n)}$ have only nonconstructability of the NCF (NCF–3) type;

$G_7 \in G_6$: GTFs $\tau^{(n)}$ define parallel global one–to–one mappings of the form $\tau^{(n)}: C(A, d) \to C(A, d)$ $(d \geq 1)$.

It can be shown [10,11] that with respect to the composition of the sets, G_k *(**k=1..6**)* form non–commutative semigroups, and the set G_7 forms a group. Thus, the semigroup ***L(a, d)*** of all parallel global mappings $\tau^{(n)}: C(A, d) \to C(A, d)$ in classical ***d–CA****s* $(d \geq 1)$ can be presented as a certain combination of a finite number of *disjoint semigroups* and a *group*, i.e. $L(a, d) = \cup_k G_k$ $(k=1..7)$. Analysis of the structures of the semigroups G_k $(k=1..6)$ and the group G_7 allowed to formulate a rather interesting result related to the *decomposition* operation of the semigroup ***L(a,d)*** of parallel mappings for classical ***d–CA****s* $(d \geq 1)$:

The semigroup L(a, d) of all parallel mappings $\tau^{(n)}: C(A, d) \to C(A, d)$ defined by the classical d–CAs $(d \geq 1)$ can be represented by combining

six disjoint subgroup G_k (k = 1..6), which do not have finite generative systems, and 1 the maximum group G(d). The sets G_h (h = 4..6) relative to the semigroup L(a, d)\G(d) are isolated subpolugroups.

Further researches have shown that the structure of the ***G(d)*** group itself remains somewhat open. A more detailed study of binary classical ***1-CAs*** to establish one-to-one mappings $G_k: C(B, 1) \rightarrow C(B, 1)$, that differ from identical ones, was rather successful. The following result presents the best for today, which is of some theoretical interest, in particular, in the formal researches of classical ***1–CAs*** [10,11,113,141,183].

Semigroup L(a, 1) of 1–dimensional global mappings $\tau^{(n)}: C(A, 1) \rightarrow C(A, 1)$, ($a \geq 3$), defined by a classical 1–CA, it is presented in the form of union of 6 in pairs nonequivalent not intersecting G_k subsemigroups (k = 1..6), which do not have finite generating systems, and 1 maximum group G(a), which is a combination of the subgroup T* of all identical mappings $\tau^{(n)}_o$ ($n \geq 2$) with finite system of generatrices P(a, 2) together with the relation $\tau^{(n)(a-1)!} = \tau^{(2)}_o$ and maybe subgroups of one–to–one mappings other than the above mappings.

That result, along with some others, suggests the need to continue studies in this direction, given the full variety of possibilities already for binary ***1-CAs***. Meantime, despite the results obtained, they do not give complete solution to the structure of even the ***G(1)*** group which participates in the specified representation of the semigroup ***L(a, 1)*** of ***1***-dimensional global mappings. In addition, from the results obtained, it follows that the group ***G(d)*** in the representation of the semigroup ***L(a, d)*** will have to contain non–trivial identical one–to–one global mappings, while for each of the six subgroup G_k (k = 1..6) defined by the representation ***L(a, d), d–PDF,*** generally speaking, will have the negative solution.

Note, based on the concept of infinite mutually erasable configurations ***(∞–MEC)***, one can define another approach to solving the decomposition problem of global transition functions in classical ***CAs***.

Two configurations $c_1, c_2 \in C(A, d, \infty)$ – couple of infinite mutual erased configurations (∞–MEC) in only case when for them the relationship

$$c_1\tau^{(n)} = c_2\tau^{(n)} = c_3 \in C(A, d, \infty) \neq \square$$

where $c_1 \neq c_2$ and `□` – completely zero configuration of space Z^d which according to the above postulate belongs to the set $C(A,d,\phi)$ takes place.

We have identified one useful class $E^{\#}$ of global transition functions $\tau^{(n)}$ of subpolugroup G_4 whose local transition functions $E^{(n)}$ will be defined below. This class of global transition functions due to specific dynamic properties is of quite certain interest in study, regardless of the ***d–PDF.***

$$E^{(n)}(x_1,x_2,...,x_n)=0,\ \text{if}\quad x_j=0,\ x_{n-1}=x_n=1\quad (j=1..n-2)$$
$$E^{(n)}(x_1,x_2,...,x_n)=1,\ \text{if}\quad x_j=0,\ x_{n-1}=1\quad (j=1..n-2,n)$$
$$E^{(n)}(x_1,x_2,...,x_n)=x_n,\ \text{otherwise}$$

Now, based on the analysis of sets of pairs ∞–***MEC*** for global transition functions from set $E^{\#}$, can be established the exact appearance of similar pairs for each function from the class $E^{\#}$, allowing to prove the result:

For an arbitrary integer $n \geq 3$, the global transition functions $E^{(n)}$ from class $E^{\#}$ have, in general, the negative 1–PDF solution.

This result allows to obtain a constructive negative ***1–PDF*** solution that have certain interesting applications. Some of them are considered in a different context [11]. On the basis of a certain class of global transition functions $\tau^{(n)}$ of G_4, that is defined in a special way and whose functions satisfy the appropriate conditions, using a rather detailed analysis of the structures of pairs ∞–***MEC*** existing for them, it is possible to formulate a result sufficiently useful for some of applications of a theoretical nature:

Subpolugroup G_4 of univariate binary global transition functions $\tau^{(n)}$ has no a finite basis.

It should be noted that the proposed method of solving ***1–PDF,*** based on the concept of ∞–***MEC,*** is a significant generalization of the method of solving the decomposition problem of global functions based on results on the nonconstructability problem in classical ***CA****s* [10,11]. We studied the structural features of the ∞–***MEC*** in sufficient detail with obtaining the accompanying results [10,11,113,141,146,161,179,182-196].

Based on an algebraic approach which is of interest to the mathematical theory of ***CA****s,* along with its many applications, we have addressed some issues of investigation of both the generalized problem ***(d–GPDF)*** and the general problem ***(d–PDF)*** of the decomposition of global transition functions. First of all, we note the fundamental difference between the general problem and the generalized one: *the generalized decomposition problem differs from the general problem in that the decomposition allows the use of global transition functions with neighborhood patterns of the same size as the original global transition function*. Moreover, both of these decomposition problems are not equivalent – for certain global transition function, ***d–GPDF*** may have a solution, while for ***d–PDF*** it is not. In [10,113,179] provides interesting examples of this type. Meanwhile, ***d–GPDF*** can well be considered as a kind of private case that in certain cases that is of interest. To analyze binary one-dimensional both classical and non-classical ***CA****s,* the ***PDF*** procedure programmed in *Maple* system was used. Calling ***PDF(m)*** returns a list of format {[*a,b*], [*c,d,g*],...,[*e, j*], [*l, s*]}, whose subscriptions with ***2*** and/or ***3*** elements will define the numbers of global transition functions that make up all valid compositions of the simpler global transition functions for the original global transition function with the given number ***m,*** if such compositions exist; otherwise, the procedure call returns the empty list. In [10,11,113] presents a number of results on the collected by the procedure statistics.

So, based on the use of a certain modification of ***PDF*** procedure, a list of the numbers of all ***1***–dimensional binary global transition functions with the neighborhood index ***X*** = {*0,1,2*}, which do not have compositions of simpler global functions is obtained. A fairly simple calculation shows that the proportion of ***CAs*** with this property is *0.76*, that is, more than *3/4*. So, of all classic one–dimensional binary ***CAs*** with neighborhood index ***X*** = {*0,1,2*}, which possess a universal reproducibility attribute in the *Moore* sense of finite configurations, only three with numbers *60, 90, 102* can be represented as compositions of two simpler functions with numbers {[*7,4*], [*4,7*], [*13,7*], [*10,13*]}, {[*7,7*], [*10,7*]}, {[*10, 11*], [*7, 6*], [*6, 7*], [*11, 7*]}, respectively. In addition, global linear transition function with the number *105*, cannot be represented in the form of a composition of two simpler global transition functions. The rather significant potential for the decomposition problem of global functions is ensured by the use of global transition functions in the same alphabet along with unrelated neighborhood indices. And first of all, this applies to the case when the decomposition problem is considered relative to the given subclasses of global transition functions, and in this regard we have obtained a number of interesting enough results. With the class of linear and strictly linear global transition functions, numerous computer experiments were carried out in terms of studying the decomposition problem, using rather simple procedures programmed in *Mathematica*. Note, that theoretical results using numerous and comprehensive computer experiments performed in *Mathematica* made it possible to formulate useful proposals. In particular, among the global functions of ***d***–dimension defined in the finite alphabet, at least ***4*** disjoint sets of global functions can be distinguished regarding the possible solution of ***d–PDF (d ≥ 1)*** [10,11,148,161,179,182-196]:

♦ *global transition functions that do not have a positive **d–PDF** solution;*

♦ *global transition functions with positive **d–PDF** solutions;*

♦ *global transition functions with a single positive **d–PDF** solution;*

♦ *global functions having a single positive **d–PDF** solution consisting of a degree of some simpler global transition function (**d ≥ 1**).*

A slightly different approach was used by us on the basis of *I. Zhegalkin* polynomials. From the theory of Boolean functions it is known that the Boolean function can be represented by the *Zhegalkin* polynomial, i.e. a binary local transition function $\sigma^{(n)}$ can be uniquely represented by the appropriate *Zhegalkin* polynomial from ***n*** variables of degree no higher than ***n.*** In a more general statement, it should be borne in mind, binary ***CAs*** are most suitable for study in the *Zhegalkin* algebra, which is a type of algebra of logic [10]. In addition, the *Zhegalkin* algebra allows a quite natural generalization in the case of ***a***–valued logics, if ***a*** is the degree of a certain prime number. This allows to use the apparatus of polynomial theory over finite fields quite efficiently to study multi-valued logics and classical ***d-CAs*** for the case of more general types of finite alphabet ***(d≥1)***. A number of discussions in this area can be found in [10,11,141,148].

In particular, based on the study of class ***G*** of local transition functions presented in the form of polynomials of a special form over the set field ***A*** together with the class of all binary local transition functions presented by *I. Zhegalkin* polynomials, the following basic result can be obtained:

For primes a and n, not every local function $\sigma^{(n)} \in G$ can be represented in the form of a superposition of a finite number of simpler functions in the same alphabet A. For each prime number n ≥ 3, the binary local transition functions $\sigma^{(n)} \in G$ cannot be represented in the form of some superposition of the finite number of simplest local transition functions $\sigma^{(j)} \in G$ in the same binary alphabet A.

From the evidence follows that, based on the polynomial representation of local transition functions $\sigma^{(n)}$ by polynomials modulo ***a,*** except for the case of the composite number ***a,*** quite it is possible to obtain constructive solutions to the decomposition problem of the global transition functions without using the concept of basis. In addition, on the basis of this, it is easy to prove the absence of a finite basis for a set of all global transition functions $\tau^{(n)}$ classical ***d–CAs (d ≥ 1)***. In this way, you can obtain some general criterion for solving the decomposition problem for an arbitrary global transition function $\tau^{(n)}$ defined in the finite alphabet $A=\{0,1,...,a-1\}$ ***(a** is a prime number)* [10,11,148,161,182]. In particular, our research in this direction gave a rather unexpected result:

The proportion of all global transition functions of d-dimension, which are defined in an arbitrary alphabet A which allow positive solutions of d–PDF and d–GPDF, is zero (d ≥ 1).

So the ***d–PDF*** study, instead of proving the existence of the negativity of its solution, turned into a search for its rather rare positive solutions. At last, from our results on the study of ***d–PDF*** and ***d–GPDF,*** it is possible to establish that among all ***d***–dimensional global transition functions $\tau^{(n)}$ $(n \geq d+1)$ defined in A_p alphabet, a certain *hierarchy of complexity* of the global transition functions $\tau^{(n)}$ with respect to the decomposition problem can be determined.

Let us say that an arbitrary global transition function $\tau^{(n)}$ belongs to the s–level of complexity [***s < n; the notation: $\tau^{(n)} \in L(s)$***] ***if and only if representations exist for the given global transition function $\tau^{(n)}$ in the following form:***

$$\tau^{(n)} = \tau_1^{\left(n_1^p\right)} \tau_2^{\left(n_2^p\right)} \tau_3^{\left(n_3^p\right)} \dots \tau_{k_p}^{\left(n_{k_p}^p\right)};\ n > d+1;\ s = min\left\{max\left\{n_1^p, \dots, n_{k_p}^p\right\} \| p = 1..m\right\}$$

that is, for a global function $\tau^{(n)}$, the PDF has a positive solution. If the PDF for the global transition function $\tau^{(n)}$ has a negative solution, then such global transition function is assigned to the complexity class L(n).

Based on our results and definitions, the following asymptotic relations can be obtained, having rather many quite important applications in the problems of classical cellular automata [10,11,148,161,183], namely:

$$(\forall s \geq 2)(\# L(s) > 0); \quad \lim_{s\to\infty} \# L(s) \Big/ a^{a^s} \geq 1$$

where ***a*** is a prime number and ***#G*** is cardinality of the finite set ***G.*** We have shown that there is the next rather important result of the solvability of the complexity levels of global transition functions in classical ***CAs***, which is primarily of theoretical interest in the study of the algorithmic properties of the dynamics of classical ***d–CAs*** as conceptual models of spatially distributed dynamical systems ***(d ≥ 1):***

The problem of determining whether an arbitrary d-dimensional global transition function $\tau^{(n)}$ given in an alphabet A belongs to the s-difficulty level (s ≤ n) is, in general, is algorithmically solvable.

Thus, based on the introduced concept of complexity for global transition functions relative to ***d–PDF (d–GPDF) (d≥1)***, you can obtain interesting characteristics of global transition functions $\tau^{(n)}$**.** It follows from results that we essentially used the algebraic properties of the finite alphabet A_p**,** since a local transition function can be unambiguously represented by a polynomial modulo ***a*** of maximum degree ***n*(a–1)*** over the field A_p**,** and vice versa. While in the case of the A_c alphabet, not every local transition function given in the alphabet of this type can be presented in polynomial form. Namely, the following main result occurs [10,11,113,183]:

For an arbitrary finite alphabet A_c = {0,1,2, ..., a–1} the fraction (W) of local transition functions $\sigma^{(n)}$, that are defined in the states alphabet of this type and which allow polynomial presentations modulo a satisfies the following relation: $1 \Big/ a^{a^n - 4^n} \leq W \leq 1 \Big/ a^{a^n - (a-2)^n}$.

It follows from the result – for the case of a composite integer ***a,*** almost all local transition functions $\sigma^{(n)}$ defined in the A_c states alphabet cannot be represented in polynomial form modulo ***a*** for sufficiently large values ***n*** and/or ***a.*** In this regard, the following question is naturally formulated: *Is it possible to determine such algebraic system in which a polynomial representation can be defined for the local transition function defined in the A_c alphabet*? As a result of the analysis, we have proposed one rather interesting example of an algebraic system, in the environment of which almost all local transition functions determined in the A_c alphabet can be uniquely represented by polynomials modulo ***a.*** Based on a number of our studies, the following rather interesting result was obtained regarding ***d–PDF*** and ***d–GPDF*** in the case of the A_c alphabet of classical ***d–CA****s;* this result is of significant theoretical interest for ***CA****s* issues in general along with a number of applications [10,11,113,141-147,183]:

Relative to almost all global transition functions determined in the A_c alphabet, whose local transition functions allow representations in the form of polynomials in the above form, both d–PDF and d–GPDF will be both equivalent and algorithmic solvable.

So, the above results of the ***d–PDF*** and ***d–GPDF*** study extend to almost all global transition functions defined in the A_c alphabet. Whereas so far we cannot extend them to the general case of alphabet ***A,*** which requires more research. Along with the above algebraic method, that are based on polynomial representations of local transition functions, methods and results of algebraic theory of ***a***–valued logics, for example, iterative *Post* algebras, can be successfully used for their formal studies [10,11]. In this regard, it is interesting to identify and research the class of certain non–traditional algebraic systems within which acceptable representations of local transition functions defined in the arbitrary alphabet ***A*** are possible. Finally, the research of various sets of global transition functions closed relative to the composition operation is of a certain interest from many standpoints. In this regard, we investigated a number of similar sets that are interesting from an applied standpoint in the context of dynamic and extreme capabilities of classical ***CA****s*. In our works, you can find many interesting examples of using some other operations on global transition functions, as well as a more detailed discussion of ***PDF/GPDF*** [10,11].

In this we conclude a certain conceptual presentation of ***TRG*** studies in the theory of mainly classical cellular automata, moving to our point of view on the formation of this scientific direction. Naturally, our point of view is to a certain extent subjective, but it is based on our many studies in the early stages of the development of the theory of cellular automata. In addition, a number of serious studies on the historical aspects of ***CA****s* are well consistent with our standpoint on this subject.

5.8. The main stages of the cellular automata theory formation

At the end of the chapter and this book as a whole, we will present our standpoint on the history of cellular automata, given our familiarity with the problematics in the early stages of its formation as a certain separate direction. Today, the problematics of cellular automata *(**CA**s, **CA**-models)* is rather well advanced, being quite independent direction of the modern mathematical cybernetics, having own terminology and axiomatic at the existence of broad enough domain of various appendices. In addition, it is necessary to note that at assimilation of this problematics in the *Soviet Union* in *Russian*-lingual terminology, whose basis for the first time have been laid by us at *1970*, for the concept *«Cellular automata»* the term *«Homogeneous structures» (**HS**s; **HS**–models)* has been determined that nowadays is the generally accepted term together with a whole series of other our notions, definitions and denotations. While a rather detailed list of publications on ***CA**s* problematics can be found, for example, in [154]. Therefore, during the present survey along with this term its well–known *Russian*–lingual equivalent *«Homogeneous structures»* can be used too.

Cellular automaton (CA) **–** a parallel information processing system that consists of infinity intercommunicating identical finite *Mealy* automata *(elementary automata)*. We can interpret ***CA**s* also as a theoretical basis of artificial high parallel information processing systems. From logical standpoint a ***CA*** is an infinite automaton with specific internal structure. So, the ***CA**s* theory can be considered as structural and dynamical theory of the infinite automata. At that, ***CA**s* can serve as an excellent basis for modeling of many discrete processes, representing interesting enough independent objects for research too. Recently, the undoubted interest to ***CA*** problematics *(above all in the applied aspect)* has arisen anew, and in this direction many remarkable results have been obtained. In addition, by ***CA**s* and ***CA*** we will mean cellular automata and a separate cellular automaton, depending on the context without causing misunderstandings.

Thus, the ***CA***–axiomatics provides ***three*** fundamental properties such as ***homogeneity, localness*** and ***parallelism*** of functioning. If in a similar computing model we shall with each elementary automaton associate a separate microprocessor then it is possible to unrestrictedly increase the sizes of similar computing system without any essential increase of its temporal and constructive expenses, required for each new expansion of the computing space, and also without any overheads connected to the coordination of functioning of an arbitrary supplementary quantity of elementary microprocessors. Similar high–parallel computing models admit practical realizations consisting of large enough number of rather elementary microprocessors which are limited not so much by certain architectural reasons as by a lot of especially economic and technologic reasons defined by the modern level of development of microelectronic technology, however with the great potentialities in the future, first of all, in light of rather intensive works in field of nanotechnology. In addition, ***CA*** models can be used successfully for problems solving of information transformation such as encryption, encoding and data compression [113].

The above three such features as *high homogeneity, high parallelism* and *locality of interactions* are provided by the ***CA***–axiomatic itself, whereas such property important from the physical standpoint as *reversibility* of dynamics is given by program way. In light of the listed properties even classical ***CAs*** are high–abstract models of the real physical world, which function in a space and time. Therefore, they in many respects better than many others formal architectures can be mapped onto a lot of physical realities in their modern understanding. Moreover the ***CA***–concept itself is enough well adapted to solution of different problems of modelling in such areas as mathematics, cybernetics, development biology, theoretical physics, computing science, discrete synergetic, dynamic systems theory, robotics, etc. Numerous visual examples available for today lead us to a conclusion that ***CAs*** can represent a rather serious interest as a new rather perspective environment of modelling and research of different discrete processes and phenomena, determined by the above properties; at that, by bringing the ***CA***–problematics on a new interdisciplinary level and, on the other hand, as a rather interesting independent formal mathematical object of researches [3,113,117,122-124,138-146,155-161,175-178,182].

The base modern tendencies of elaboration of perspective architecture of high parallel computer facilities, the problem of modelling of discrete parallel processes, discrete mathematics and synergetic, theory of parallel discrete dynamical systems, problems of artificial intellect and robotics, parallel information processing and algorithms, physical and biological modelling, along with a lot of other important prerequisites in different areas of modern natural sciences define at the latest years a new ascent of the interest to the formal cellular models of various type which possess high parallel manner of acting; the cellular automata are some of major models of such type. During time which has passed after appearance of the first monographs and the collected papers which have been devoted to various theoretic and applied aspects of the ***CAs*** problems, the certain progress has been reached in this direction, that is connected, above all, with successes of theoretical character along with essential expansion of field of appendices of the ***CA***–models, especially, in computer science, cybernetics, physics, modelling, developmental biology and substantial growth of number of researchers in this direction. At the same time in the *USA, Japan, Germany, the Great Britain, Hungary, Estonia*, etc., a series of works summarizing the results of progress in those or other directions of ***CAs*** problematics including its numerous appendices in various fields has appeared. Our monographs and reports at a certain substantial level have represented the reviews of the basic results received by the *Tallinn Research Group (**TRG**)* on the ***CA*** problematics and its application [113]. From the very outset of our researches on the ***CA*** problematics, above all, with an application accent onto mathematical developmental biology the informal ***TRG*** consisting of researchers of some leading scientific centres of the former *USSR* has gradually been formed up. At that, the ***TRG*** staff was not strictly permanent and was being changed in rather broad bounds depending on the researched problems. In works [113,140-148,155-161] the analysis of the ***TRG*** activity instructive to some degree for research of the dynamics of development of the ***CA*** problematics as a independent scientific direction as a whole had been represented. Ibidem, our basic directions of research can be found along with main received results.

Today, cellular automata are being investigated from many standpoints and interrelations of objects of such type with already existing problems are being discovered constantly. For purposes of general acquaintance with extensive ***CAs*** problematics as a whole along with its separate basic directions specifically, we recommend to address oneself to interesting and versatile reviews of such researchers as *V.Z. Aladjev, V. Cimagalli, K. Culik, D. Hiebeler, A. Lindenmayer, A. Smith, P. Sarkar, T. Toffoli, M. Mitchell, R. Vollmar, S. Wolfram,* et al. [154]. A series of books and monographs of the authors such as *V.Z. Aladjev, T. Toffoli, R. Vollmar, A. Adamatzky, E. Codd, A. Ilachinskii, M. Garzon, M. Duff, P. Kendall, B. Voorhees, M. Sipper, O. Martin, K. Preston, S. Wolfram, N. Margolus, B. Voorhees, V. Kudrjvcev,* and some others contain a rather interesting historical excursus in the ***CA*** problematics; at that, unfortunately, hitherto a common standpoint onto historical aspect in this question is absent. In view of that, here is a rather opportune possibility to briefly emphasize once again our standpoint on historical aspect of the ***CAs*** problematics: a brief historical excursus presented below make it one's aim to define the basic stages of becoming of the ***CAs*** problematics, having digressed from numerous particulars [113,141-144,163,175-178,183].

Having started own study on the ***CA***-problematics in *1969*, we on base of analysis of large number of publications and direct dialogue with many leading researchers in this direction have a quite certain information that concerns the objective development of its basic directions, above all, of theoretical character. That allows us with sufficient degree of objectivity to note the pivotal stages of its development; at the same time, numerous details of historical character concerning the ***CA***–problematics the reader can find, for example, in a whole series of works presented in links [154].

From theoretical standpoint the ***CAs*** concept *(Homogeneous structures)* has been introduced at the end of the forties of the past century by *John von Neumann* on *S. Ulam's* advice with purpose of determination of more realistic and well formalized model for research of behaviour of complex evolutionary systems, including self–reproduction of the alive organisms. Whereas *S. Ulam* has used ***CA***–like models, in particular, for researches of the growth problem of crystals and certain other discrete systems that grows in conformity with recurrent rules. The structures which have been investigated by him and his colleagues were, mainly, of dimensionality ***1*** and ***2,*** however higher dimensions have been considered too. In addition, questions of universal computability along with certain other theoretical questions of behaviour of cellular structures of such type also were kept in view. A little bit later also *A. Church* started to investigate the similar structures in connection with works in field of infinite abstract automata and mathematical logic [154]. *J. von Neumann's* ***CA***–model has received the further development in works of him direct followers whose results with the finished and edited work of the first one have been published by *A.W. Burks* in his excellent works [154], which in many respects have determined development of researches in the given direction for several subsequent years. In process of researches on the ***CA***–problematics *A.W. Burks* has organized at the Michigan university the research team *«The Logic of Computer Group»*, from which a whole series of the first–class experts on the ***CA***–problematics has come out afterwards *(J. Holland, R. Laing, T. Toffoli, and many others)*.

At the same time, considering historical aspect of the ***CA***–problematics, we should not forget an important contribution to the given problematic which was made by pioneer works *Konrad Zuse (Germany)* and with which the world scientific community has been familiarized enough late and even frequently without his mention in this historical aspect. At that, *K. Zuse* not only has created the first programmable computers *(1935–1941)*, has invented the first high–level programming language *(1945)*, but was also the first who has introduced idea of *«Rechnender Raum» (Computable Spaces)*, or in the modern terminology – *Cellular Automata*. Furthermore, *K. Zuse* has supposed that physical processes in point of fact are calculations, while our universe is a certain *«cellular automaton»* [154]. In the late seventies of the last century such view on the universe was innovative, whereas now the idea of the computing universe horrify nobody, finding logical place in the modern theories of some researchers working in the field of quantum mechanics [154]. Unfortunately, even at present the *K. Zuse's* ideas are unfamiliar to rather meticulous researchers in this field. For exclusion of any speculative historical aspects existing occasionally today, in the following historical researches it is necessary to pay the most steadfast attention on this rather essential circumstance. So, namely therefore, only many years later the similar ideas have been republished, popularized and redeveloped in research of other researchers such as *S. Wolfram, T. Toffoli, E. Fredkin*, et al. [3,113,154]. In addition, the ***CA**s* concept itself has been entered by *John von Neumann*. Perhaps, *John Neumann*, being familiar with *K. Zuse* ideas, could apply cellular automata not only for simulation of process of reproducing automata, but also for creation of high parallel computing model, but it did not happen.

From more practical standpoint and game experiment the ***CA***-models has notified about itself in the late *sixties* of the last century when *J. Conway* has presented the now known game *«Life»*. This game became a rather popular and has attracted attention to cellular automata of both numerous scientists from different fields and amateurs [154]. At the same time, this game, probably, is the most known ***CA*** model; at that, it will possess the ability to self–reproduction and universal computing. By modelling the process of work of an arbitrary *Turing* machine by means of a ***CA*** model, *J.H. Conway* has proved ability of the model to universal computability. Later a rather simple manner of implementation of any Boolean function in configurations of the *«Life»* has been suggested [154]. So, even such simple ***CA*** model turned out equivalent to the universal *Turing* machine. Furthermore, to the given ***CA*** model the significant interest exists and till now does not disappear above all to its various computer simulating [113, 154]. Thus, early ideas and research of the first–rate mathematicians and cyberneticians such as *K. Zuse, John von Neumann, S. Ulam, A. Church* along with their certain direct followers we with good reason can ascribe to the ***first*** stage of formation of the ***CA**s* problematics as a whole.

The necessity for a good formalized media for modelling of processes of biological development and above all of self–reproduction process was being as one of the base prerequisites that stimulated the ***CA***–concept beginning. Thereupon, *John Neumann* and a whole series of his direct followers have investigated a series of questions of computational and constructive opportunities of the first ***CA***–models. The above works at the end of the fifties of the last century have attracted to the problematic a lot of researchers [154]. At that, *homogeneous structures* were being rediscovered not once and under various names: in electrical engineering they are known as iterative networks, in pure mathematics as a section of topological dynamics, in biological sciences as cellular structures, etc.

As ***second*** stage in formation of the ***CA***–problematics it is quite possible to consider publication of the widely known works of *E.F. Moore* and *J. Myhill* on the nonconstructability problem in classical ***CA***–models which along with solution of certain mathematical problems in a certain sense became accelerators of activity, attracting a rather steadfast attention to this problematics of a lot of mathematicians and researchers from other fields [154]. So, for example, we have familiarized oneself with the ***CA***–problematics in *1969* owing to *Russian* translation of the excellent work edited by *R. Bellman* that contained well–known articles of *E.F. Moore, S. Ulam* and *J. Myhill* [1]. Scientific groups on the ***CA***–problematics in the *USA, Germany, Japan, Hungary, Italy, France, and USSR (ESSR,* ***TRG,*** *1970)* are formed up. The further development and popularization of the ***CA***–problematics can be connected with names of researchers such as *E.F. Codd, S. Cole, E.F. Moore, J. Myhill, H. Yamada, S. Amoroso, E. Banks, J. Buttler, V.Z. Aladjev, J. Holland, G.T. Herman, A.R. Smith, T. Yaku, A. Maruoka, Y. Kobuchi, G. Hedlund, M. Kimura, A. Waksman, H. Nishio, T. Ostrand,* and a whole series of others researchers whose works in the *sixties* – the *seventies* of the last century have attracted attention to the given problematics from the theoretical standpoint; they have solved and formulated a lot of interesting enough problems [154]. In the future, mathematicians, physicists, and biologists began to use the ***CAs*** with the purpose of research of own specific problems. In particular, in the early *sixties* – the late *seventies* of the last century the numerous researchers have prepared entry of the ***CA***–problematics into the current stage of its development that is characterized by join of earlier disconnected ideas and methods on the general conceptual and methodological platforms, along with a rather essential expansion of fields of its application.

We can attribute the beginning of the ***third*** period to the early *eighties* of the last century when to ***CA***–problematics the special interest again has been renewed in connection with rather active researches on the problem of artificial intellect, physical modelling, elaboration of a new perspective architecture of high–parallel computer systems, and a lot of important motivations. So, in our opinion namely since the works of the researchers such as *Bennet C., Grassberger P., Boghosian B., Crutchfield J., Chopard B., Culik II K., Gács P., Green D., Gutowitz H., Langton C., Martin O., Ibarra O., Kobuchi Y., Margolus M., Mazoyer J., Toffoli T., Wolfram S., Aladjev V., Bandman O.,* etc. a new splash of interest to the ***CAs*** began as a perspective environment, above all, of physical modelling. A rather extensive selection of references, including references on both the *Soviet* and the *Russian–language* authors, can be found in [154]. So, at present, ***CA***–problematics are being rather widely studied from extremely various standpoints and interrelations of similar homogeneous structures with existing problems are constantly sought and discovered. A lot of rather large teams of researchers in many countries and first of all in the *USA, Germany, the Great Britain, Italy, France, Japan, Australia* deals with this problematics. Scientific activity in this direction was carried out and in *Estonia* within of the ***TRG*** whose a whole series of results has received an international recognition and has made up essential enough part of a fairly developed modern ***CA***–problematics.

The modern standpoint on the ***CA**s (**HS**s)* theory has been formed under the influence of works of researchers such as *Adamatzky A.I., Aladjev V., Amoroso S., Arbib M., Bagnoli F., Bandini S., Bandman O.L., Bays C., Banks E.R., Barca D., Barzdin J., Binder P., Boghosian B., Bolotov A.A., Burks A.W., Butler J., Cattaneo G., Chate H., Chowdhury D., Church A., Codd E.F., Crutchfield J.P., Culik K.II, Das A.K., Durand B., Durret R., Fokas A.S., Fredkin E., Gács P., Gardner M., Gerhardt M., Griffeath D., Golze U., Grassberger P., Green D., Gutowitz H.A,, Hedlund G., Honda N., Cole S., Hemmerling A., Holland J., Ibarra O., Ikaunieks E., Jen E., Ilachinskii A., Kaneko K., Kari J., Kimura M., Kobuchi Y., Kudryavtsev V.B, Langton C., Legendi T., Lieblein E., Lindenmayer A., Maneville P., Margolus N., Martin O., Maruoka A., Mazoyer J., Mitchell M., Moore E.F., Morita K., Myhill J., Nasu M., Neumann J., Nishio H., Ostrand T., Pedersen J., Podkolzin A., Sato T., Richardson D., Sarkar P., Sipper M., Smith A., Shereshevsky M., Sutner K., Takahashi H., Thatcher J., Toffoli T., Toom A.L., Tseitlin G.E., Varshavsky V.I., Vichniac G., Vollmar R., Voorhees B., Wuensche A.A., Waksman A., Weimar J., Willson S., Yaku T. Wolfram S.,* and other numerous researchers from many countries.

Along with our works in the ***CA***-problematics, it is necessary to note a lot of Soviet researchers who have received in the field the fundamental and rather considerable results at the *sixties* – the *eighties* of the last century. Here they: *Adamatzky A.I. (identification of **CA**s models), Bandman O.L. (asynchronous **CA**s), Blishun A. (growth of patterns), Bliumin S. (growth of patterns), Bolotov A.A. (simulation among classes of **CA**s), Varshavsky V.I. (synchronization of **CA**s, simulation of anisotropic **CA**s on isotropic ones), Georgadze A., Mandzhgaladze P., Matevosian A. (growth of the configurations; universal stochastic and deterministic **CA**s, **CA**–models and parallel grammars), Dobrushin R.L., Vasil'ev N., Stavskaya O.N., Mitiushin L., Leontovich A., Toom A.L., (probabilistic **CA**s), Ikaunieks E. (nonconstructible configurations), Koganov A.V. (universal **CA**s, stable configurations, simulation of **CA**s), Kolotov A.T. (the models of excitable media), Levenshtein V. (synchronization in **CA**s), Kurdiumov G.L. and Levin L.A. (stochastic **CA**s), Makarevskii A.I. (implementation of Boolean functions in **CA**s), Petrov E. (synchronization of **2d–CA**s), Podkolzin A.S. (simulation of **CA**s; asymptotic of the global dynamic; universal **CA**s), Pospelov D. (homogeneous structures and distributed AI in **CA**–models), Evreinov E., Prangishvili I. (**CA**-architecture of high-parallel processors), Reshod'ko L. (**CA**s of excitable media), Revin O. (simulation of anisotropic **CA**s on isotropic **CA**-models), Solntzev S. (growth of patterns), Tzetlin M.*

(collectives of automata, games in the ***CA****s), Tzeitlin G. (algebras of shift registers), Scherbakov E.S. (universal algebras of parallel substitutions)*, and a whole series of others domestic researchers [113,176-178,183].

It is supposed that the ***CA***–models can play extremely important part as both conceptual and the applied models of spatially–distributed dynamic systems among which first of all an especial interest the computational, physical and biological cellular systems present. In the given direction already takes place a rather essential activity of a lot of the researchers who have received quite encouraging results [154]. At last, theoretical results of the above–mentioned and of a lot of other researchers have initiated a modern mathematical ***CAs*** theory evolved to the current time into an independent branch of the abstract automata theory that has a rather numerous interesting appendices in various areas of science and technics, in particular, in fields such as physics, developmental biology, parallel information processing, creation of perspective architecture of high–efficiency computer systems, computing sciences and informatics, which are linked to mathematical and computer modelling, etc., and by substantially raising the ***CA***–concept onto a new interdisciplinary level. Our concise enough standpoint on the main stages of development and formation of the ***CA*** theory is given above; for today there is a number of the reviews devoted to this question, for example [154], many works on the ***CA***–problematics in varying degree concern this question also [154]. Furthermore, it should be noted that the matter to a certain extent has a rather subjective character, and that needs to be meant.

Meanwhile, the separate researchers in a gust of certain euphoria try to represent the ***CA***–approach as an universal remedy of the solution of all problems and knowledge of outward things, identifying it with a ***«****New kind****»*** of science of universal character. In this connection it is necessary to mark the vast and pretentious book of *S. Wolfram* [162], whose title has rather advertising and commercial, than scientific–based character. This book contains many results that have been obtained much earlier by a lot of other researches on ***CA***-problematics, including the *Soviet* authors *(see references in* [154] *and some others)*. At the same time, the priority of many fundamental results in this field belongs to other researchers. The unhealthy vanity of the author of the book does not allow him to look without bias on history of the ***CA***–problematics as a whole. In general, *S. Wolfram* enough frivolously addresses with authorship of the results that were received in ***CA***–problematics, therefore there can be a impression – everything made in this field belongs basically to him. At that, the book contains basically results of computer modelling with very simple types of ***CA***–models, drawing the conclusions and assumptions on their basis with rather doubtful reliability and quality. In the book we can meet an irritating density of passages in which the author takes personal credit for ideas that are *«common knowledge»* among experts in the relevant fields. Seems, such *S. Wolfram* passages and inferences similar to them cause utterly certain doubts in scientific decency and judiciousness of their author. At last, we absolutely do not agree that *Wolfram* book presents a *"New kind"* of science; nevertheless his book would be more pleasant to read if he were more modest. In our opinion, the given book represents in many respects a speculative sight both on ***CA***–problematics, and on the science as a whole. Here we only shall note, contrary to the pursued purposes the book not only was not revelation for the researches working in the ***CA***–problematics but also to a certain extent has caused a little bit deformed representation about the research domain which is perspective enough from many points of view. With relatively detailed point of view that concerns the book, the reader can familiarize in works [113,154] and some others. Meanwhile, in spite of the told above relative to the book, it can represent a certain interest, taking into consideration the marked and some other remarks. In our opinion, the *Wolfram* book doesn't introduce of anything essentially new in the cellular automata theory above all in its mathematical component, wearing, rather, a certain claim character, but this is already on the author's conscience; in fact, the given book is most likely of a scientific and popularization nature and nothing more.

At last, we will make one essential enough remark concerning of place of the ***CA***–problematics in scientific structure. By synchronization with the standpoint on ***CA***–problematics that is declared by our books [157-161] a vision of the given question is being presented as follows. Our long–term experience of investigations in the ***CA***–problematics both on theoretical and especially applied level speaks entirely about another, namely:

(1) CA*–models (cellular automata, homogeneous structures) represent a special class of infinite abstract automata with specific internal structure that provides extremely high–parallel level of the information processing and calculations; these models form a specific class of discrete dynamic systems which function in especially parallel way on base of principle of local short–range interaction;*

(2) CA *can serve as a satisfactory model of high–parallel processing just as Turing machines (Markov normal algorithms, productions systems, Post machines, etc.) serve as formal models of sequential calculations; from this point of view the* ***CA****–models it is possible to consider and as algebraic processing systems of finite or infinite words, defined in finite alphabets, on the basis of a finite set of rules of parallel substitutions; in particular, a* ***CA****–model can be interpreted as a certain system of parallel programming where the rules of parallel substitutions act as a parallel language of the lowest level programming;*

*(**3**) the principle of local interaction of elementary automata composing a **CA**–model that in result defines their global dynamics allows to use the **CA** and as a fine environment of modelling of a rather broad range of the processes, phenomena and objects; furthermore, the phenomenon of the reversibility permitted by the **CA**s does their by interesting enough means for physical modelling, and for creation of rather perspective computing structures basing on the nanotechnologies;*
*(**4**) **CA**–models represent a rather interesting independent mathematical object whose essence consists in high–parallel processing of words both in finite and infinite alphabets.*

At that, it is possible to associate the ***CA***–approach with a certain model analogue of the differential equations in partial derivatives that describe those or another processes with that difference, if differential equations describe a process at the average, in a ***CA***–model defined in appropriate way, a certain researched process is really embedded and dynamics of the ***CA***–models enough evidently represents the qualitative behaviour of researched process. Thus, it is necessary to determine for an elementary automaton of the model the necessary properties and rules of their local interaction by appropriate way. The ***CA***–approach can be used for study of processes described by complex differential equations which have not of analytical solution, and for the processes that cannot be described by such equations. Moreover, the ***CA*** present a rather perspective modelling environment for research of those phenomena, processes, and objects for which there are no known classical means or they are complex enough.

As we already noted, as against many other modern fields of science, the theoretical part of the ***CA***–problematics is no so appreciably crossed with its *second* applied component, therefore we can consider ***CAs*** problems as two independent enough directions: study ***CA*** as mathematical objects and use ***CAs*** for modelling; at that, the *second* direction is characterized even by the wider spectrum. The level of evolution of the ***2nd*** direction is appreciably being determined by possibilities of the modern computing systems since ***CA***–models, as a rule, are being designed on base of the immense number of elementary automata and, as a rule, with complex enough rules of local interaction among themselves.

The indubitable interest to them amplifies also a possibility of practical realization of high parallel computing ***CAs*** on basis of modern successes of microelectronics and prospects of the information processing at the molecular level *(methods of nanotechnology);* whereas the ***CA***–concept itself provides creation of conceptual and practical models of different spatially–distributed dynamic systems of which namely physical systems are the most interesting and perspective. Indeed, models that in obvious way reduce macroscopic processes to rigorously determined microscopic processes represent especial epistemological and methodical interest for they possess the great persuasiveness and transparency. Namely, of this standpoint the ***CA***–models of different type represent a special interest, above all, from the applied standpoint at research of a lot of phenomena, processes and objects in various fields and, first of all, in developmental biology, physics and computer science.

The *first* direction enough intensively is developed by mathematicians whereas contribution to development of the *second* direction essentially more representative circle of researchers from various theoretical and applied fields *(physics, chemistry, biology, technique, etc.)* brings. Thus, if theoretical researches on the ***CA***–problematics in general are limited to the classical, polygenic and stochastic ***CA***–models, then the results of the *second* direction are based on essentially wider representation of classes and types of ***CA***–models. As a whole, if classical ***CA***–models represent first of all the formal mathematical systems researched in the appropriate context, then their numerous generalizations represent a rather perspective environment of modelling of various processes and objects.

In the conclusion once again it is necessary to note an important enough circumstance, at discussion of the *Classical cellular automata (**CCA**)* we emphasize the following a rather essential moment. We considered the ***CCA***–models which are a class of parallel discrete dynamic systems as certain formal algebraic systems of processing of finite configurations *(words)* in finite alphabets whatever, as a rule, to their microprogrammed environment, i.e. without use of their cellular organization on the lowest level inherent into them, what distinguishes our approach to research of the given objects from approaches of a lot of other researchers. Also, we consider ***CCA***–models as a formal mathematical object having specific inside organization without ascribing to them a certain universality and generality in perception of the *World*. At similar approach the ***CCA*** are considered at especially formal level not allowing in full measure to use their intrinsic property of high parallelism in field of computations, and information processing as a whole.

Naturally, for solution of a lot of different applied problems in the ***CA***–environment and obtaining of a series of thin results first of all of model character an approach on microprogram level is needed when a studied process, algorithm or phenomenon is directly embedded in ***CA***–models, using its parameters: *dimension, neighbourhood index*, a *states alphabet* and a *local transition function*. At such approach it is possible to receive solutions of a lot of important appendices with generalizations of a rather high level of theoretical character. In particular, by direct embedding of universal computing algorithms or logical elements into such objects it is possible to constructively prove existence of the universal computability, etc. In spite of such extremely simple concept of the ***CCAs***, they by and large have a rather complex dynamics. In many cases theoretical research of their dynamics collides with a rather complexity. Therefore, computer simulation of these structures which in empirical way allows to research their dynamics is a rather powerful tool. For this reason this question is quite natural for investigations of the ***CA***–problematics, considering the fact that ***CA***–models at the formal level present the dynamical systems of high–parallel substitutions.

Indeed, the problem of computer modelling of the ***CAs*** is solved at two main levels*: (**1**) simulation of **CA**s dynamics on computers of traditional architecture*, and (***2****) simulation on the hardware architecture which as much as possible corresponds to the **CA**s concept; so-called **CA**-oriented architecture of computing systems*. Thus, computer simulation of the ***CA***-models plays a rather essential part at theoretical study of their dynamics; meanwhile, it is even more important at practical realizations of the ***CAs*** models of different processes. At present, a whole series of interesting systems of software and hardware for help of investigations of different types of the ***CA***–models have been developed; their characteristics can be found in the references [154]. In our works a lot of programs in various program systems for different computer platforms had been presented. Among them a lot of interesting programs for simulation of ***CA***–models in the *Mathematica* and *Maple* systems has been programmed. On the basis of computer simulation many of interesting theoretical results on the ***CCA*** and their use in the fields such as mathematics, developmental biology, computer sciences, etc. had been received. However, the given matter along with applied aspects of the ***CA***–models in the present book aren`t considered, despatching the interested reader to a rather detailed discussion of these aspects to the corresponding publications in lists of references [154] and in references given in [155-161]; a lot of interesting works can be found in *Internet* by the corresponding key phrases too.

The problematics considered by the ***TRG*** study in many respects has been conditioned by interests and tastes of the authors along with traditions of creative activity of the ***TRG*** in this field. At last, we will note that in our activity it is possible to allocate ***3*** main directions: *(**1**) study of classical **CA**s as a formal parallel algorithm of processing of configurations in the finite alphabets, (**2**) applications of the classical and the generalized **CA**s in mathematics and computer facilities of highly parallel action*, and *(**3**) mathematical and developmental biology*. With our main results in ***2*** last directions the interested reader can familiarize in sufficient detail in [113, 141,155-161,182-196] and in numerous references contained in them.

CONCLUSION

In the present book, we have attempted to summarize the main directions and results of the creative activity of the *Tallinn Research Group (**TRG**)* since its inception *(1970)* in such fields as *cellular automata* – the root cause of the formation of the *Group*, mainframes, personal computers, programming, automated control systems, computing sciences, computer mathematics systems *(MathCAD, Maple, Mathematica)*, general statistics theory and others. It cannot be said that this book is unique in this field, ***TRG*** and has previously periodically published reports on its scientifical and practical activities, for instance in *Moscow* for the periods *1969-1993* and *1995–1998*, but they were of a slightly different nature and they did not consider a number of important activity directions of the *Group*.

Along with this, ***TRG*** members were done both individual lectures and their courses in the above fields in universities and organizations in such countries as *Estonia, Lithuania, Belarus, Ukraine, Russia* and certain others. Within the same fields, ***TRG*** members have prepared and issued many publications of both monographic and textbooks along with a lot of periodicals. Our publications cover such countries as the *USSR, Estonia, Belarus, Hungary, Russia, Germany, Lithuania, Czechoslovakia, Japan, Ukraine, the USA, Holland, Bulgaria, Great Britain*, etc. A number of these publications were popular enough, quoted, and a number of them were posted on the *Internet* for free. Along with these original editions, we developed a large library *UserLib* of new software tools *(more than 850 tools)*, won the *Smart Award* network award, and a unified package *MathToolBox (more 1420 tools)* for *Maple* and *Mathematica* systems, respectively, expanding the functionality of these systems. Working on these popular tools was carried out not only in the process of preparing the corresponding textbooks for *Maple* and *Mathematica* systems, but also as a result of a very detailed testing of both systems. In *2018*, both software tools were posted on the *Internet* in free access [183].

The presented material allows to see in historical retrospect our tortuous enough path, from interest in cellular automata, fundamental fields to the purely applied fields, and vice versa. At the same time, the choice of a particular area was formed under the influence of both scientific interests and the interests of a particular department, with which the members of the ***TRG*** had close production relations. At last, to summarize the above, it should be noted that the presented book is both summing and final in nature, due to the fact that the most active members of the ***TRG*** and the *Baltic branch* of the *International Academy of Noosphere* are at a rather serious age, not stimulating a serious creative activity of the both *Groups*.

REFERENCES

1. Mathematical problems in the biological sciences // Proc. of symposia in applied mathematics. Ed. *R. Bellman:* AMS: N.Y, v. 14, p. 250, 1962.
2. Myhill J. The converse to Moore`s garden–of–Eden theorem // Proc. Amer. Math. Soc., 14 (1963), pp. 685–686.
3. Aladjev V.Z. To the Theory of the Homogeneous Structures.– Tallinn: Estonian Academic Press, 1972, 259 p. *(in Russian with English)*
4. Aladjev V.Z., Tupalo V.G. Scientific and practical activity of the *TRG: The final results for the 25–th anniversary* (*1969–1993*).– Moscow: Russian Academy of Cosmonautics, 1994, 115 p., ISBN 5942003948.
5. Aladjev V.Z., Hunt Ü.J., Shishakov M.L. Research Activity of the *TRG: Scientific Report for Period 1995–1998.*– Tallinn: TRG Press, 1998, 80 p., ISBN 1406429856.
6. Aladjev V.Z., Osipov O.B. Introduction to the *UCS (the Unified Computers System)* operating system. v. *1, 2*.–Tallinn: Soyuztorgsystem, 1975 *(in Russian with extended English summary)*
7. Aladjev V.Z., Osipov O.B. Introduction to the architecture of the *UCS* models.–Tallinn: Valgus Press, 332 p., 1976 *(Russian with English)*
8. Mathematical support of *UCS* and *ACS* / Ed. *V.Z. Aladjev*.– Tallinn: Valgus Press, 190 p., 1978 *(in Russian with English summary)*
9. Database management system based on the operating system *MINIOS* and DBMS *OKA* / Ed. *V.Z. Aladjev*.– Tallinn: Valgus Press, 180 p., 1980
10. Parallel information processing and parallel algorithms / Ed. *V.Z. Aladjev*.– Tallinn: Valgus Press, 298 p., 1981 *(in Russian and English)*
11. Parallel information processing systems: *UCS* software materials // Ed. *V.Z. Aladjev*.– Tallinn: Valgus Press, 376 p., 1983 *(in Russian with extended English summary)*
12. Aladjev V.Z. Description of a parallel processing system for series *IBM/360* in terms of system of algorithmic algebras // Proc. All–Union seminar on parallel programming and high–productivity computational systems.– Kiev, 1982, pp. 75-87 *(in Russian with English summary)*
13. Aladjev V.Z., Osipov O.B. An approach to creation of *HPC* on basis of hardware of *CUCC* // Computational Systems, 3, 1983 *(in Russian)*
14. Aladjev V.Z. A configuration of operating system *OS/360* for *ACS* of Ministry of Trade of the *ESSR* // Proc. VNIIET–Systems.– Moscow, 1977, pp. 80-87 *(in Russian with extended English summary)*
15. Aladjev V.Z. Software industry, in [8], pp. 145–156 *(in Russian)*
16. Aladjev V.Z., Fishman J.I. An approach to creation of control program of *ACS* software on the basis of *OS/360*, in [8], pp. 128–144 *(in Russian with extended English summary)*

17. Aladjev V.Z., Klushnikov V.V. Comparative analysis of operating systems *DOS/360* and *OS/360*, in [8], pp. 118–127 *(in Russian)*
18. Aladjev V.Z., Osipov O.B. Training of problem programmers for the third generation computers, in [8], pp. 104–117 *(in Russian)*
19. Aladjev V.Z., Lukash N.A., Mazurenko V.I. Problems of realization of the homogeneous computational systems on the basis of *IBM/360*–compatible computers, in [10], pp. 8–166 *(in Russian)*
20. Aladjev V.Z., Blinder B.S., Kuznecov V.V. An approach to creating of high–productivity processing systems on the basis of technique of the *CUCC* // Proc. All–Union conference on trends, techniques and problems in modern software.– Sevastopol, 1981 *(in Russian)*
21. Aladjev V.Z., Blinder B.S., Mazurenko V.I. An approach to creating of high–productivity information systems on the basis of information paralleling in homogeneous computational systems // Proc. All–Union symp. on high–productivity databases and inform. systems.– Kishinew, 1982 *(in Russian with extended English summary)*
22. Aladjev V.Z., Boiko V.K. Information paralleling in homogeneous computer systems // Proc. All–Union Conf. of VGPTI.– Tallinn, 1982, pp. 42–99 *(in Russian with extended English summary)*
23. Aladjev V.Z. Lectures on personal computers *WANG 2200MVP*.– Tallinn: SKB MPCM *ESSR*, 1986, 180 p. *(in Russian)*
24. Aladjev V.Z. and et al. Personal computer *ISKRA–226:* Architecture and software.– Kiev: Ukrainian Soviet Encyclopaedia, 1989, 150 p. *(in Russian with extended English summary)*
25. Aladjev V.Z., Sirodza I.B. Solving engineering problems in the operating environment of the language *Basic* of PC *ISKRA–226*.– Kharkov: Kharkov University, 1988, 120 p. *(in Russian with English)*
26. Aladjev V.Z. and et al. Design of software for PC *ISKRA–226*.– Kiev: Technika Press, 1989, 255 p., ISBN 5335002409 *(in Russian)*
27. Aladjev V.Z., Shilenko V.F. Personal computer *ISKRA–1030:* Architecture and software.– Kiev: Ukrainian Soviet Encyclopaedia, 1990, 484 p., ISBN 5885000174 *(in Russian with English summary)*
28. Aladjev V.Z. Measuring of liquids and gases expense by method of standard diaphragms // CNTII–Project of Gosstroja *USSR*.– Moscow, 1988, 180 p. *(in Russian with extended English summary)*
29. Aladjev V.Z., Prokopovich M.I., Lebedev E.V. Utilities for personal computers.– Kiev: Technika Press, 1992, 152 p., ISBN 5335011947 *(in Russian with extended English summary)*
30. Aladjev V.Z., Tupalo V.G. Scientific editing and statistics on personal computers.– Moscow: Mintopenergo Press, 1993, 106 p. *(in Russian with extended English summary)*

31. Aladjev V.Z., Tupalo V.G. *Turbo–Pascal* for All.– Kiev: Technika Press, 1993, 176 p., ISBN 5335012250 *(in Russian with English)*
32. Aladjev V.Z., Tupalo V.G. Computer Telecommunication.- Moscow: Mintopenergo Press, 1993, 140 p. *(in Russian with English summary)*
33. Aladjev V.Z., Tupalo V. Computer Chrestomathy.– Kiev: Ukrainian Soviet Encyclopaedia, 1993, 448 p., ISBN 5885000500 *(in Russian)*
34. Aladjev V.Z., Hunt Ü., Shishakov M.L. Fundamentals of informatics: *Textbook for university students*. ***2nd*** updated and expanded edition.– Moscow: Filin Press, 1999, 542 p., ISBN 589568131X *(in Russian)*
35. Aladjev V.Z., Hunt Ü., Shishakov M.L. Fundamentals of computer informatics.– Belarus: Gomel: TRG Press, 1997, 396 p., ISBN 5140642545 *(in Russian with extended English summary)*
36. Aladjev V.Z., Haritonov V.N. General Theory of Statistics Course.– Palo Alto: Fultus Books, 2006, 250 p., ISBN 1596820861 *(in Russian with extended English summary)*
37. Aladjev V.Z., Hunt Ü., Shishakov M.L. General Theory of Statistics Course.– Belarus: Gomel: BELGUT Press, 1995, 201 p. *(in Russian)*
38. Aladjev V.Z., Veetõusme R.A., Hunt Ü. General theory of statistics.– Tallinn: TRG & Salcombe Press, 1995, 200 p., ISBN 1995146428 *(in Russian with extended English summary)*
39. Aladjev V.Z., Haritonov V.N. General Theory of Statistics.– USA: Palo Alto: Fultus Books, 2004, 256 p., ISBN 1596820128.
40. Aladjev V.Z., Gershgorn N.A. Computational problems on personal computer.– Kiev: Technika Press, 1991, 248 p., ISBN 5335010649 *(in Russian with extended English summary)*
41. Aladjev V.Z., Tupalo V.G. Algebraic computations on computers.– Moscow: Mintopenergo Press, 1993, 252 p., ISBN 5942004568 *(in Russian with extended English summary)*
42. Aladjev V.Z., Hunt Ü.J., Shishakov M.L. Mathematics on a personal computer.– Gomel: BELGUT Press, 1996, 498 p., ISBN 34206140233.
43. Aladjev V.Z., Shishakov M.L. Introduction into mathematical package *Mathematica 2.2*.– Moscow: Filin Press, 1997, 363 p., ISBN 5895680046 *(in Russian with extended English summary)*
44. Aladjev V.Z., Vaganov V.A., Hunt Ü., Shishakov M.L. Introduction into environment of mathematical package *Maple V*.– Belarus: Minsk: International Academy of Noosphere, the Baltic Branch, 1998, 452 p., ISBN 1406425698 *(in Russian with extended English summary)*
45. Aladjev V.Z., Vaganov V.A., Hunt Ü., Shishakov M.L. Programming in environment of mathematical package *Maple **V***.– Minsk–Moscow: Russian Ecology Academy, 1999, 470 p., ISBN 4101212982 *(in Russian)*
46. Aladjev V.Z., Bogdevicius M.A. Solution of physical, technical and

mathematical problems with *Maple V*.– Tallinn–Vilnius, TRG, 1999, 686 p., ISBN 9986053986 *(in Russian with extended English summary)*
47. Aladjev V.Z., Vaganov V.A., Hunt Ü., Shishakov M.L. Workstation for the mathematicians.– Tallinn–Minsk–Moscow: Russian Academy of Natural Sciences, 1999, 608 p., ISBN 3420614023 *(in Russian)*
48. Aladjev V.Z., Shishakov M.L. Workstation of the mathematician.– Moscow: Laboratory of Basic Knowledge, 2000, 752 p. + CD, ISBN 5932080523 *(in Russian with extended English summary)*
49. Aladjev V.Z., Bogdevicius M.A. *Maple* ***6:*** *Solution of mathematical, statistical, engineering and physical problems*.– Moscow: Laboratory of Basic Knowledge, 2001, 850 p. + CD, ISBN 593308085X *(in Russian)*
50. Aladjev V.Z., Bogdevicius M.A. Special questions of operation in environment of the mathematical package *Maple*.– Vilnius: Vilnius Gediminas Technical University, 2001, 208 p. + CD with Library, ISBN 9985927729 *(in Russian with extended English summary)*
51. Aladjev V.Z., Bogdevicius M.A. Interactive ***Maple:*** Solution of mathematical, statistical, engineering and physical problems.– Tallinn: International Academy of Noosphere, the Baltic Branch, 2001–2002, CD with Booklet, ISBN 9985927710.
52. Aladjev V.Z., Vaganov V.A., Grishin E.P. Additional software of mathematical package *Maple* of releases ***6*** and ***7***.– Tallinn: International Academy of Noosphere, the Baltic Branch, 2002, 314 p. + CD with our Library, ISBN 9985927737 *(in Russian with extended English summary)*
53. Aladjev V.Z. Effective operation in mathematical package *Maple*.– Moscow: BINOM Press, 2002, 334 p. + CD, ISBN 593208118X *(in Russian with extended English summary)*
54. Aladjev V.Z., Liopo V.A., Nikitin A.V. Mathematical package *Maple* in physical modeling.– Grodno: Grodno State University, 2002, 416 p., ISBN 3093318313 *(in Russian with extended English summary)*
55. Aladjev V.Z., Bogdevicius M.A., Prentkovskis O.V. A New software for mathematical package *Maple* of releases ***6, 7*** and ***8***.– Vilnius: Vilnius Gediminas Technical University, 2002, 404 p., ISBN 9986055652 *(in Russian with extended English summary)*
56. Aladjev V.Z., Vaganov V.A. Systems of computer algebra: *A new software toolbox for Maple*.– Estonia: Tallinn: International Academy of Noosphere, the Baltic Branch, 2003, 270 p. + CD, ISBN 9985927761.
57. Aladjev V.Z., Bogdevicius M.A., Vaganov V.A. Systems of computer algebra: *A new software toolbox for Maple*. *Second* edition.– Estonia: Tallinn: International Academy of Noosphere, 2004, 462 p., ISBN 9985927788 *(in Russian with extended English summary)*
58. Aladjev V.Z. Computer Algebra Systems: *A New Software Toolbox*

for Maple.– Palo Alto: Fultus Books, 2004, 575 p., ISBN 1596820004.
59. Aladjev V.Z., Bogdevicius M.A. *Maple:* Programming, Physical and Engineering Problems.– USA: Palo Alto: Fultus Books, 2006, 404 p., ISBN 1596820802, Adobe Acrobat eBook *(pdf)*, ISBN 1596820810.
60. Aladjev V.Z. Computer Algebra Systems. *Maple: Art of Programming*.– Moscow: BINOM Press, 2006, 792 p., ISBN 5932081899 *(in Russian with extended English summary)*
61. Aladjev V.Z. Foundations of programming in *Maple:* Textbook.– Tallinn: International Academy of Noosphere, 2006, 300 p., *(pdf)*, ISBN 998595081X, 9789985950814 *(in Russian with English summary)*
62. Aladjev V.Z., Boiko V.K., Rovba E.A. Programming and applications elaboration in *Maple: Monograph*.– Belarus: Grodno: Grsu, 2007, 456 p., ISBN 9789854178912 *(in Russian and English summary)*
63. Aladjev V.Z., Vaganov V.A. Modular programming: *Mathematica* vs *Maple, and vice versa*.– USA, CA: Palo Alto, Fultus Books, 2011, 418 p. ISBN 9781596822689.
64. Aladjev V.Z., Bezrukavyi A.S., Haritonov V.N., Hodakov V.E. Programming: System *Maple* or *Mathematica*?– Ukraine: Kherson, Oldi–Plus Press, 2011, 474 p., ISBN 9789662393460.
65. Aladjev V.Z., Boiko V.K., Rovba E.A. Programming in systems *Mathematica* and *Maple: A Comparative Aspect*.– Belarus: Grodno, Grodno State Univ., 2011, 517 p., *(in Russian with English summary)*
66. Aladjev V.Z., Grinn D.S., Vaganov V.A. The extended functional tools for system *Mathematica*.– Ukraine: Kherson: Oldi–Plus Press, 2012, 404 p., ISBN 9789662393590 *(in Russian with English summary)*
67. Aladjev V.Z., Grinn D.S. Extension of functional environment of system *Mathematica*.– Ukraine: Kherson: Oldi–Plus Press, 2012, 552 p., ISBN 9789662393729 *(in Russian with extended English summary)*
68. Aladjev V.Z., Grinn D.S., Vaganov V.A. The selected system problems in software environment of system *Mathematica*.– Ukraine: Kherson: Oldi–Plus Press, 2013, 556 p., ISBN 9789662393729.
69. Aladjev V.Z., Vaganov V.A. Extension of the *Mathematica* system functionality.– USA: Seattle, CreateSpace, An Amazon.com Company, 2015, 590 p., ISBN 9781514237823.
70. Aladjev V.Z., Vaganov V.A. Toolbox for the *Mathematica* programmers.– USA: Seattle, CreateSpace, An Amazon.com Company, 2016, 630 p., ISBN–13: 9781532748837, ISBN–10: 1532748833.
71. Aladjev V.Z., Boiko V.K., Shishakov M.L. The Art of programming in the *Mathematica* system.– Estonia: Tallinn, TRG Press, 2016, 735 p., ISBN 9789985950890 *(in Russian with extended English summary)*
72. Aladjev V.Z., Boiko V.K., Shishakov M.L. The Art of programming

in the *Mathematica* system. Second edition.– USA: Raleigh, Lulu Press, 2016, 735 p., ISBN 9781365560736.
73. Aladjev V.Z., Shishakov M.L. Software etudes in the *Mathematica*.– USA: CreateSpace, 2017, 614 p., ISBN 1979037272.
74. Aladjev V.Z., Bogdevicius M.A. *Employment of package Maple V for solution of physical and technical problems* // Proc. Intern. Conferen. *TRANSBALTICA–99*, April 1999, Lithuania, Vilnius.
75. Aladjev V.Z., Hunt Ü.J. *A workstation for mathematicians* // Proc. Intern. Conferen. *TRANSBALTICA–99*, April 1999, Lithuania, Vilnius.
76. Aladjev V.Z., Hunt Ü.J. *A workstation for mathematicians* // Proc. Intern. Conf. Improvement of Cont. Mechanism, Belarus, Grodno 1999.
77. Aladjev V.Z., Shishakov M.L. *Programming in package Maple V* // Intern. Conf. *"Computer algebra in fundamental and applied researches and education"*, Belarus, Minsk, 1999.
78. Aladjev V.Z., Shishakov M.L. *Automated working place of the mathematician* // Intern. Conf. *"Computer algebra in fundamental and applied researches and education"*, Belarus, Minsk, 1999.
79. Aladjev V.Z., Bogdevicius M.A. *Application of the package Maple V for solution of physical and technical problems* // Intern. Conference *TRANSBALTICA–99*, Lithuania, Vilnius, VGTU, 1999.
80. Aladjev V.Z., Hunt Ü.J. *A workstation for mathematician* // Intern. Conf. *TRANSBALTICA–99*, Lithuania, Vilnius, VGTU Press, 1999.
81. Aladjev V.Z., Hunt Ü.J. *A workstation for mathematician* // Intern. Conf. *"Perfecting of mechan. of management"*, Belarus, Grodno, 1999.
82. Aladjev V.Z., Shishakov M.L. *Programming in package Maple V* // ***2nd*** Intern. Conference *"Computer algebra in fundamental and applied researches and education"*, Belarus, Minsk, 1999.
83. Aladjev V.Z., Shishakov M.L. *Automated working place of the mathematician* // ***2nd*** Intern. Conf. *"Computer algebra in fundamental and applied researches and education"*, Belarus, Minsk, 1999.
84. Aladjev V.Z., Shishakov M.L., Trohova T.A. *Educational computer laboratory of the engineer* // Proc of the ***8th*** Byelorussian mathematical conference, vol. *3*, Belarus, Minsk, 2000.
85. Aladjev V.Z., Shishakov M.L., Trohova T.A. *Modelling in programme environment of mathematical package Maple V* // Proc. of Intern. Conf. on Mathematical Modelling, Ukraine, Kherson, 2000.
86. Aladjev V.Z., Shishakov M.L., Trokhova T.A. *Workstation for the solution of systems of differential equations* // Third Intern. Conference *"Differential Equations and Applications"*, St.–Petersburg, 2000.
87. Aladjev V.Z., Shishakov M., Trokhova T. *Computer laboratory for engineering researches* // Intern. Conf. *ACA–2000*, St.–Petersburg, 2000.

88. Aladjev V.Z., Bogdevicius M.A., Hunt Ü.J. *A workstation for mathematicians* // Lithuanian conference *"TRANSPORT–2000"*, Lithuania, Vilnius: Technika Press, 2000, pp. 274–282.
89. Aladjev V.Z. *Computer algebra system Maple**:** A new software library* // Proc of Intern. Conf. *ICCS 2003*, LNCS 2657, Springer, 2003, pp. 711–717, ISBN 9783540448600.
90. Aladjev V.Z., Boiko V.K. *Extension of computer mathematics systems Maple and Mathematica* // Intern. scientific and practical conf. *«ITES–2015»*. Intern. Seminar–school of Mathem. Modelling in *CAS «KAZCAS–2015»*, Kazan, 2015, pp. 96–112, ISBN 9785969002692.
91. Aladjev V.Z., Boiko V.K. *Tools for computer research of cellular automata dynamics* // Intern. science seminar *«Nonlinear models in the mechanics, statistics, theory field and cosmology»*. Intern. scientific and practical conf. *«ITES–2017»*. Intern. seminar of math. modelling in *CAS «KAZCAS–2017»*, Kazan, 2017, pp. 7–15, ISBN 9785969003897.
92. Aladjev V.Z., Shishakov M.L., Vaganov V. Practical programming in *Mathematica*. ***5th*** edition.– USA: Raleigh, Lulu Press, 2017, 613 p., ISBN 9781387356065 *(in Russian with English summary)*
93. Aladjev V.Z., Shishakov M.L., Vaganov V.A. Functional and procedural programming in *Mathematica*.– USA: KDP Press, 2020, 352 p., ISBN 9798650293118.
94. Aladjev V.Z., Shishakov M.L., Vaganov V.A. *Mathematica**:*** Functional and procedural programming. ***2nd*** edition.– USA: KDP Press, 2020, 396 p., ISBN 9798574872710.
95. Aladjev V.Z., Vaganov V.A., Hunt Ü., Shishakov M.L. Introduction to mathematical package *Maple V*.– Belarus: Minsk: IAN Press, 1998, 452 p., ISBN 1406425698 *(in Russian with English summary)*
96. Aladjev V.Z., Vaganov V.A., Hunt Ü.J., Shishakov M.L. Programming in the environment of mathematical package *Maple V*.– Belarus: Gomel: Salcombe Press, 1999, 470 p., ISBN 4101212982 *(in Russian with extended English summary)*
97. Aladjev V.Z., Vaganov V.A., Hunt Ü.J., Shishakov M.L. A workstation for mathematicians.– Tallinn–Gomel–Moscow: VASCO & Salcombe, 1999, 608 p., ISBN 3420614023 *(in Russian)*
98. Aladjev V.Z., Shishakov M.L. Automated workplace of mathematician.– Moscow: Basic Knowledge Laboratory, 2000, 752 p., ISBN 5932080523 *(in Russian with extended English summary)*
99. Aladjev V.Z., Bogdevicius M.A. Solution of physical, technical and mathematical problems with *Maple V*.– Vilnius: Technics Press, 1999, 686 p., ISBN 9986053986p *(in Russian with extended English summary)*
100. Aladjev V.Z., Shishakov M.L. Workstation of the mathematician.–

Moscow: Laboratory of Basic Knowledge, 2000, 752 p. + CD, ISBN 5932080523 *(in Russian with extended English summary)*
101. Aladjev V.Z., Bogdevicius M.A. *Maple **6:*** Solution of statistical, mathematical and engineering–physical problems.– M.: Laboratory of Basic Knowledge, 2001, 850 p. + CD, ISBN 593308085X *(in Russian with extended English summary)*
102. Aladjev V.Z., Bogdevicius M.A. Special questions of operation in environment of the mathematical package *Maple*.– Vilnius: Vilnius Gediminas Technical University, 2001, 208 p. + diskette with Library, ISBN 9985927729 *(in Russian with extended English summary)*
103. Aladjev V.Z. Additional software for the mathematical package *Maple* of releases ***6*** and ***7*.**– M.: Laboratory of Basic Knowledge, 2002, 334 p. + CD, ISBN 593208118X *(in Russian with English summary)*
104. Aladjev V.Z., Liopo V.A., Nikitin A.V. Mathematical package *Maple* in physical modelling.– Grodno: Grodno State University, 2002, 416 p., ISBN 3093318313 *(in Russian with extended English summary)*
105. Aladjev V.Z., Vaganov V.A. Computer Algebra System *Maple**:** A new software library*.– Tallinn: International Academy of Noosphere, the Baltic Branch, 2002, CD with Booklet, ISBN 9985927753.
106. Aladjev V.Z., Bogdevicius M.A., Prentkovskis O.V. New software for mathematical package *Maple* of releases ***6, 7*** and ***8*.**– Vilnius: Vilnius Gediminas Technical Univ., 2002, 404 p., ISBN 9985927745.
107. Aladjev V.Z., Vaganov V.A. Systems of Computer Algebra: *A new software toolbox for Maple*.– Tallinn: International Academy of Noosphere, the Baltic Branch, 2003, 270 p. + CD, ISBN 9985927761.
108. Aladjev V.Z., Bogdevicius M.A., Vaganov V.A. Systems of Computer Algebra: *A new software toolbox for Maple*. *Second* edition.– Tallinn: International Academy of Noosphere, the Baltic Branch, 2004, 462 p., ISBN 9985927788.
109. Aladjev V.Z. Computer Algebra Systems: *A new software toolbox for Maple*.– Palo Alto: Fultus Books, 2004, 575 p., ISBN 1596820004.
110. Aladjev V.Z., Bogdevicius M.A. *Maple:* Programming of physical and engineering problems.– USA: Palo Alto: Fultus Books, 2006, 404 p., ISBN 1596820802.
111. Aladjev V.Z., Boiko V.K., Rovba E.A. Application programming and development in *Maple*.– Belarus: Grodno: GrSu Press, 2007, 458 p., ISBN 9789854178912 *(in Russian with extended English summary)*
112. Aladjev V.Z., Boiko V.K., Rovba E.A. Programming in *Maple* and *Mathematica* packages: *Comparative aspect*.– Belarus: Grodno: GrSu Press, 2011, 517 p., ISBN 9789855154816 *(in Russian)*
113. Aladjev V.Z. et al. Electronic library of books and software for

scientists, experts, teachers and students in natural and social sciences.– USA: Palo Alto: Fultus Books, 2005, CD, ISBN 1596820136.
114. Aladjev V.Z. Computer Algebra System *Maple: A new software Library* // Proc. Intern. Conf. Comp. Sciences – ICCS 2003, Australia, Melbourne, 2003, pp. 711-717.
115. Aladjev V., Bezrykavy V., Haritonov V., Hodakov V. Programming: *Maple* or *Mathematica**?** Comparative aspect*.– Ukraine: Kherson, Oldi–Plus Press, 2011, 476 p., ISBN 9789662393460.
116. Bukusheva A.V. Organization of students' independent work in studying computer geometry in *LMS MOODLE* // Azimuth of scientific research: *Pedagogy and psychology*, vol. 5, no. 3(16), 2016.
117. Aladjev V.Z., Baskin V.E., Belousov L.V. et al. Mathematical Developmental Biology.– Moscow: Science Press, 1982, 256 p.
118. Aladjev V.Z. Two models solving the French Flag Problem // Proc. AN *ESSR* Fiz.-Math., 20, no. 3, 1971 *(in Russian with English summary)*
119. Aladjev V.Z., Orav T.A. A numerical model of regulation of axial structures // Proc. AN *ESSR*. Fiz.-Mathem., 20, no. 3, 1971 *(in Russian)*
120. Aladjev V.Z., Poluektov R.F. About models of regulation of multi–dimensional structures // Proc. AN *ESSR*. Biology, 20, no. 4, 1971 *(in Russian with extended English summary)*
121. Aladjev V.Z. Some algorithmic questions in mathematical biology of development // Proc. AN *ESSR*. Biology, 22, no. 1, 1973 *(in Russian with extended English summary)*
122. Aladjev V.Z. Survey of researches in the theory of homogeneous structures and their applications // Math. Biosci., 22, 1974, pp. 121-154.
123. Aladjev V.Z. The behavioral properties of homogeneous structures // Math. Biosciences, 24, 1976, pp. 224-242.
124. Aladjev V.Z. Some new results in the theory of homogeneous structures // Intern. Symp. on Math. Topics in Biology, Kyoto, 1978.
125. Aladjev V.Z. A problem about matrices, arizing in the theory of self–reproducing automata // Proc. AN *ESSR*. Fiz.-Mathem., 19, no. 2, 1970 *(in Russian with extended English summary)*
126. Aladjev V.Z. Some questions, arizing in the theory of homogeneous structures // Proc. of AN *ESSR*. Biology, 19, no. 3, 1970.
127. Aladjev V.Z., Orav T.A. Problems of the cybernetical modelling of the development processes // Proc. AN *ESSR*. Biology, 23, no. 3, 1974.
128. Aladjev V.Z. Cybernetical modelling of the development biology, in collection [11], pp. 211-280 *(in Russian with English summary)*
129. Lindermayer A. Developmental algorithms for multi–cellular organisms: *A survey of L–systems* // J. Theor. Biol., 1975, v. 54, p. 3-22.
130. Lindenmayer A. Algorithms for plant morphogenesis // Theoretical

plant morphology.– The Hague: Leiden Univ. Press, 1978, pp. 38-81.
131. Lindenmayer A., Culik K. Growing cellular systems: Generation of graph by parallel rewriting // Intern. J. Gen. Syst., 1981.
132. Rosenberg G., Penttonen M., Salomaa A. Bibliography of ***L***–system // Theor. and Comp. Sci., 1977, v. 1, no. 5, pp. 339-354.
133. Herman G., Rosenberg G. Developmental systems and languages.– Amsterdam: North–Holland, 1975.
134. Apter M. A formal model of biological development // Proc. AN *ESSR*. Fiz.-Mathem., 22, no. 3, 1973, pp. 233-245.
135. Lück G., Lück J. Automata theoretical explanation of tissue growth // Proc. Intern. Symp. on mathematical topics in biology, Kyoto, 1978.
136. Aladjev V.Z. *Tau(n)*–grammars and generated by them languages // Proc. AN *ESSR*. Biology, 23, no. 1, 1974, pp. 67-87.
137. Aladjev V.Z. Operations on languages generated by *Tau(n)*–grammars // Com. Mathem., Univ. of Carolinae Praga, 15, no. 2, 1974.
138. Aladjev V.Z. On the equivalence of *Tau(n)*–grammars and *Sb(m)*–grammars // Com. Mathem., Univ. of Carolinae Praga, 15, no. 4, 1974.
139. Aladjev V.Z. Cybernetic modelling of developmental biology / Parallel information processing and parallel algorithms.– Tallinn: Valgus Press, 1981, pp. 138–191 *(in Russian with English summary)*
140. Aladjev V.Z. Selected questions of the theory of homogeneous structure / Parallel algorithms and parallel computing software.– Tallinn: Valgus Press, 1982, pp. 112–162 *(in Russian with English summary)*
141. Aladjev V.Z. Survey of research in theory of classical homogeneous structures and their appendices // Tech. Rept., no. 18–1–12/88 *(revised and extended report)*.– Tallinn: Project–Technological Institute of the Industry, 1989, 2087 p. *(in Russian with extended English summary)*
142. Aladjev V.Z. Solutions of a number of problems in mathematical theory of homogeneous structures, TR–040684.– Tallinn: P/A Silikaat, 1985, 655 p. *(in Russian with extended English summary)*
143. Aladjev V.Z., Zinkevich T.G. Classical homogeneous structures, TR–041285.– Tallinn: P/A Silikaat, 1985, 425 p. *(in Russian with extended English summary)*
144. Aladjev V.Z. Recent results in the theory of homogeneous structures, TR–042385.– Tallinn: P/A Silikaat, 1985, 897 p. *(in Russian)*
145. Aladjev V.Z. Theoretical and applied aspects of the homogeneous structures // Means and Methods of Analogous and Digital Processing.– Tallinn: Estonian Academy of Sciences, 1988, pp. 42-57 *(in Russian)*
146. Aladjev V.Z., Hunt Ü.J. Fundamental problems in the theory of the classical homogeneous structures / TRG Research Rept. 55/97.– Tallinn: VASCO & Salcombe Eesti Ltd., 1997, 855 p. *(in Russian with extended*

English summary)
147. Aladjev V.Z. Survey on the homogeneous structures / Tech. Rept., no. 18–12/89, Tallinn: Project–Tech. Inst. of Industry, 1989, 547 p.
148. Aladjev V.Z. Classical Cellular Automata: *Mathematical Theory and Applications*.– Germany: Saarbrücken: Scholar`s Press, 2014, 517 p.
149. Aladjev V.Z. A solution of the *Steinhays's* combinatorical problem // Appl. Math. Letters, no. 1, pp. 11-12, 1988.
150. Aladjev V.Z. An algebraic system for polynomial representation of K–valued logical functions // Appl. Math. Letters, 3, pp. 207-209, 1988.
151. Aladjev V.Z. To stability of some optimal differential systems // Automat. and Comp. Technique.– Moscow: AN *USSR*, 14, 1971.
152. Aladjev V.Z. A problem about matrices arizing in the theory of the self–reproducing automata // Proc. AN *ESSR*. Fiz.-Math., 19, no. 2, 1970
153. Aladjev V.Z. Computability in the homogeneous structures, v. 1–2, 2850, 3023.– Moscow: VINITI Press, 1971, 120 p. *(in Russian)*
154. *Links on **CA**s* http://www.hs-ca.narod.ru *or* http://ca-hs.weebly.com
155. Aladjev V.Z. Mathematical Theory of Homogeneous Structures and Their Applications.– Estonia: Tallinn: Valgus Press, 1980, 267 p.
156. Aladjev V.Z. Homogeneous Structures: *Theoretical and Applied Aspects*.– Ukraine: Kiev: Technika Press, 1990, 272 p. *(in Russian)*
157. Aladjev V.Z. Classical homogeneous structures: *Cellular automata:* USA: Fultus Books, 2009, 535 p., ISBN 159682137X *(in Russian with extended English summary)*
158. Aladjev V.Z. Classical cellular automata: *Mathematical theory and applications:* Germany: Saarbrucken: Scholar`s Press, 2014, 512 p.
159. Aladjev V.Z., Shishakov M.L., Vaganov V.A. Selected problems in the theory of classical cellular automata: USA: Lulu Press, 2018, 410 p.
160. Aladjev V.Z., Shishakov M.L., Vaganov V.A. Selected problems in theory of classical cellular automata.– USA: CreateSpace, 2018, 410 p.
161. Aladjev V.Z., Shishakov M.L., Vaganov V.A. Basic elements of cellular automata theory.– USA: Lulu Press, 2019, 418 p., ISBN 9780359735129 *(in Russian with extended English summary)*
162. Wolfram S. A New Kind of Science.– N.Y.: Wolfram Media, 2002.
163. https://sites.google.com/view/cellular-automata-history/home
164. *Links on the **CA**s problematics* – http://www.hs-ca.narod.ru *or* http://ca-hs.weebly.com
165. Aladjev V.Z. Characterization of one–dimensional homogeneous structures in terms of stack automata, in [7], pp. 244–276.
166. Maruoka A., Kimura M. Condition for injectivity of global maps for tessellation automata // Information and Control, 32, 1976, pp. 158–164.

167. Kari J. Decision problems concerning cellular automata.– Finland: Turku, University of Turku, 1990.
168. Ulam S. A collection of mathematical problems.– N.Y.: Interscience Publisher, 1960.
169. Ulam S. On some mathematical problems connected with pattern of growth of figures // Proc. Symp. in Applied Mathematics, 14, 1962.
170. Aladjev V.Z. A problem about matrices, arising in the theory of the self–reproducing automata // Proc. of Academy Sciences of Estonia. Phys.–Math., 19, no. 2, 1970.
171. A library for *Maple* – https://sites.google.com/view/userlib6789
172. Aladjev V.Z. To Theory of Homogeneous Structures.– Moscow: VINITI Press, 1971, 288 p. *(in Russian with extended English summary)*
173. Aladjev V.Z. Computability in Homogeneous Structures.– Moscow: VINITI Press, 1971, 465 p. *(in Russian with extended English summary)*
174. Poupet V. Simulating *3D* Cellular Automata with *2D* Cellular Automata // LNCS 3153.– Berlin: Springer–Verlag, 2004.
175. Aladjev V.Z. The main stages of the cellular automata formation // Cybersecurity issues.– Moscow, no. 1 (47), 2022, pp. 10–17.
176. Shalyto A.A. Works on homogeneous structures and cellular automata performed in the *USSR*, *Russia* and the former republics of the *USSR* (www.computer-museum.ru/articles/books/1066/, *in Russian)*
177. Matyushkin I., Zapletina M. Overview of cellular automata based on modern domestic publications // Computer researches and modelling, no. 1, v. 11, 2019 (www.crm.ics.org.ru/journal/article/references/2764/).
178. Patrakeev I. Geospatial technologies in urban systems modelling.– Harkov, Harkov Univ., 2014, 208 p., ISBN 9789666953394 *(in Russian)*
179. Parallel algorithms and parallel computing software / ed. *V. Aladjev*. Tallinn: Valgus Press, 1982 *(in Russian with English summary)*
180. Encyclopaedia of Mathematics, vol. 1.– M.: Soviet Encyclopedia, 1977, pp. 50-53 *(in Russian)*.
181. Encyclopaedia of Physical Science and Technology, vol. *3*.– N.Y.: Academic Press, 1992, pp. 15-44.
182. Aladjev V.Z., Grinn D.S., Vaganov V.A. Classical Homogeneous Structures: *Mathematical Theory and Applications*.– Ukraine: Kherson: Oldi–Plus, 2014, 520 p., ISBN 978–966–289–035–8.

In the monograph we represent some results of the research we have done in the theory of classical *Homogeneous structures (HS–models)* and their appendices during *1969–2013*, in truth with rather considerable pauses. Meantime, during the book along with the given term its well–known Anglo–lingual equivalent «*Cellular automata* – ***CAs***» is used too. These results at present form a rather essential constituent of the ***CAs***–problematics.

183. The https://sites.google.com/view/aladjevbookssoft/home web–site contains free books in *English* and *Russian* along with software created under the guidance of the main author prof. *V.Z. Aladjev* in such areas as general theory of statistics, theory of cellular automata, programming in *Maple* and *Mathematica* systems. Each book is archived, including its cover and book block in *pdf*–format. The software with freeware license is designed for *Maple* and *Mathematica* systems.
184. Aladjev V.Z., Hunt Ü., Shishakov M.L. Questions of mathematical theory of the classical homogeneous structures.– Gomel: BELGUT Press, 1997, 151 p. *(in Russian with extended English summary)*.
185. Aladjev V.Z., Hunt Ü.J., Shishakov M.L. Mathematical Theory of Classical Homogeneous Structures.– Gomel: TRG&VASCO&Salcombe Eesti Ltd., 1998, 380 p. *(in Russian with extended English summary)*.
186. Aladyev V.Z., Boiko V.K., Rovba E.A. Classical homogeneous structures: *Theory and applications*.– Belarus: Grodno: Publishing House GrGU, 2008, 488 p., ISBN 9789985950845.
187. Aladjev V.Z. New results in the theory of homogeneous structures // MTA Szamitas es. Autom. Tanulman., 158, Budapest, 1984.
188. Aladjev V.Z. Few results in the theory of homogeneous structures // Mathem. Res., Band 25.– Berlin: Akademie–Verlag, 1985.
189. Aladjev V.Z. Unsolved Theoretical Problems in Homogeneous Structures // Math. Res., Band 48.– Berlin: Akademie–Verlag, 1988.
190. Aladjev V.Z. Survey of Some Theoretical Results and Applicability Aspects in Parallel Computation Modelling // J. of New Gener. Comput. Systems, 1, no. 4, 1988, pp. 307–317.
191. Aladjev V.Z., et.al. Unsolved theoretical problems in homogeneous structures // Proc. of the 4-th Intern. Workshop on Parallel Processing by Cellular Automata and Arrays, 342, Springer–Verlag, 1988, pp. 33–50.
192. Aladjev V.Z. Certain recent results on the theory of homogeneous structures // New Approaches to Parallel Processing.– Berlin: Akademie–Verlag, 1988, pp. 128–147.
193. Aladjev V.Z. and et al. Theoretical and Applied Aspects of the Homogeneous Structures // Proc. Intern. Workshop *PARCELLA*–90.– Berlin: Akademie–Verlag Press, 1990, pp. 48–70.
194. Aladjev V.Z. Recent Results in the Theory of Homogeneous Structures // Parallel Processing by Cellular Automata and Arrays.– Amsterdam: North–Holland Press, 1987, pp. 31–48.
195. Aladjev V.Z. Recent results in the mathematical theory of the homogeneous structures / New Trends in Comp. Sci.– Amsterdam, 1988.
196. https://all-andorra.com/viktor-aladjev-on-the-homogeneous-structures-basic-elements-and-the-cellular-automata-theory/

ABOUT THE AUTHOR

Aladjev Victor Zakharovich* –** President of the Baltic Branch of the *International Academy of Noosphere,* Prof., DSc in Mathematics. He was born on *June 14, 1942* in *Grodno (West Belarus).* In *1959* he entered the first year of the Physics and Mathematics Faculty of the University of *Grodno*, and in *1962* he was transferred to the Mathematics Department of the University of *Tartu (Estonia)*, which he successfully graduated in *1966* with a degree in *Mathematics*. In *1969 Aladjev* entered the graduate school of the Academy of Sciences of the *ESSR* with a degree in *Theory of Probability and Mathematical Statistics*, which successfully graduated in *1972* in two specialties at once, *Theoretical Cybernetics* and *Technical Cybernetics*. In *1972* he was awarded a doctorate in mathematics *(DSc)* from Prof. *R. Bellman (USA)* for his work *"Mathematical Theory of the Homogenious Structures and Their Applications";* the given work was recognized as the best in the Academy Sciences of the *ESSR* at *1972*. Since *1970, Aladjev V.* **–** President of the *Tallinn Research Group (**TRG**)* organized by him, whose scientific results subsequently have received certain international recognition, primarily in the field of researches on the mathematical theory of *homogeneous structures (Cellular Automata)*. *Cellular automata (**CA**s)*, initially being the root cause of the formation of the ***TRG, over time became a less significant field of scientifical and applied activity of the *Group*, inferior to more priority fields, in particular computer mathematics systems, although quite often interest in the ***CAs*** problematics arose again for particular periods of duration; however, the intensity of researches in this direction also decreased over time.

In *1972*, *V. Aladyev* published the first monograph on the homogeneous structures theory in the *USSR*, which was recognized as one of the best monographical publication of the Estonian Academy of Sciences in the same year, and in *1977* was noted in *Soviet Mathematical Encyclopedia* [180] and in *Encyclopaedia of Physical Science and Technology* [181]. The monograph not only presented a number of original results on this problematics, but introduced the basic *Russian*–language terminology on cellular automata too, which is now generally accepted.

From *1972* up to *1990*, he held senior positions *(chief engineer, deputy director for science)* in a number of design, technological and research organizations in *Tallinn (Estonia)*. *Aladjev's* activities at these posts were repeatedly awarded and prizes by the Council of Ministers of the *USSR*, the Central statistics Committee of the *USSR*, the All–Union Project and Tecnological Institute of the Central Statistics Committee of the *USSR*, etc. Prof. *V.Z. Aladjev* is the basic author more than *500* scientific and scientific and technical works *(including 90 monographs, textbooks and collections of articles)* published in the *USSR, Russia, Germany, Belarus, Estonia, Lithuania, Ukraine, the GDR, Czechoslovakia, Hungary, Japan, the USA, Holland, Bulgaria and Great Britain*. Since *1972*, he is referent and member of editorial board of the international mathematical journal "*Zentralblatt für Mathematik"* and since *1980*, he is a member of *IAMM (International Accociation on Mathematical Modelling)*. Prof. *V. Aladjev* is a member of the editorial boards of a number of scientific journals. He created the *Estonian School* for the mathematical theory of homogeneous structures, whose fundamental results received international recognition and have made certain contributions in the basis of a new division of the modern mathematical cybernetics. A lot of applied works of *Aladjev V.Z.* refers to computer science among which it is worth noting widely known textbooks on computer mathematics systems. Along with these original editions, he developed a large *UserLib6789* library of new software tools *(more than 850)* for which he was won the *Smart Award* network award, and a large unified *MathToolBox* package *(more 1420 tools)* for *Maple* and *Mathematica* systems, respectively, which rather essentially expand the functionality of these systems. During the preparation of these books and creation of software tools for the *Maple* and *Mathematica* systems, a sufficiently wide range of the proposals for organization, functioning and set of standard tools that improve both systems was registered, certain of which were subsequently included in subsequent versions of the systems.

In a number of fields *(mathematics, computer science, cellular automata, mathematical packages, etc.) Aladjev V.Z.* collaborates with a number of universities in the *CIS* under the program *"Visiting Professor"*. As a part of the given program and in the process of preparing a series of books on *Maple* and *Mathematica* systems, *Aladjev V.* for a lot of years gave cycles of lectures on these systems for students, graduate students and teachers of the universities of the *Baltic States* and *Belarus*, which became rather famous in the above States. In May *2015, V.Z. Aladjev* was awarded the Gold Medal *"European Quality"* by the European Scientific & Industrial Consortium *(ESIC)*, awarded for outstanding achievements in the field of science, education, production and business. From *1976* to *1990*, Dr. *V.Z. Aladjev* took an active part in the annual summer republican Olympics of state institutions of the *ESSR*, winning a number of medals in athletics. The most significant results of *V.Z. Aladjev* relate to the mathematical theory of *Homogeneous Structures (Cellular Automata)* and computer mathematics systems. *V.Z. Aladjev`s* fields of scientific interest include mathematics, cybernetics, computer science, biology, and certain other natural science fields along with preparation of books, textbooks, lectures and articles in these and related fields.