COMPUTER
METHODS
IN MATERIALS
SCIENCE

2023, vol. 23, no. 4, 57–80
https://doi.org/10.7494/cmms.2023.4.0820

Exploration of cellular automata: a comprehensive review of dynamic modeling across biology, computer and materials science

Oleksii O. Vodka*, Mariia I. Shapovalova

National Technical University "Kharkiv Polytechnic Institute", 2, Kyrpychova Str., Kharkiv, 61002, Ukraine.

Abstract

This paper delves into the expansive world of cellular automata (CA), abstract models of computation comprised of cells that interact based on predefined rules. Originating from John von Neumann's work in the 1940s, CA has evolved into a multidisciplinary field with applications ranging from mathematical concepts to complex simulations of biological, physical, computer science, material science, and social systems. The paper reviews its historical development, emphasizing John Conway's influential *Game of Life* and Burk's seminar collection. The authors categorize and explore a myriad of CA topics, including self-replicating automata, the universality of computation, compromises in CA, variants, applications in biological systems, fault-tolerant computation, pattern recognition, CA games, fractals, dynamic properties, complexity, image processing, cryptography, bioinformatics, materials modeling, probabilistic automata, and contemporary research. The significance of cellular automata for materials modeling cannot be overstated and considerable attention has been devoted to the issues of modeling nucleation and recrystallization. The review aims to provide a comprehensive resource for both beginners and experts in the field, shedding light on cellular automata's dynamic and diverse applications in various aspects of life and scientific inquiry.

Keywords: cellular automata, classification, nucleation modeling, pattern recognition, image processing, cryptography, recrystallization

Introduction

Cellular automata (CA) are abstract models of computation consisting of a network of cells, each of which can exist in a specific state. These cells interact with their nearest neighbors according to predefined rules. Each cell changes its state based on both its current state and the states of its neighbors.

The fundamental components of a cellular automaton include:

- *Cell State*: each cell can take one of a finite set of states;
- *Grid*: cells are organized into a spatial network, which can be one-dimensional, two-dimensional, or even higher-dimensional, depending on the specific task;
- *Transition Rules*: define how cell states change based on their current state and the state of their neighbors. These rules determine the evolution of the system over time.

The progenitor of the emergence and development of cellular automata is John von Neumann (1966). In the late 1940s, they were created as formal models of

*Corresponding author: oleksii.vodka@gmail.com

ORCID ID's: 0000-0002-4462-9869 (O.O. Vodka), 0000-0002-4771-7485 (M.I. Shapovalova)
Scopus author ID's: 56239259600 (O.O. Vodka), 57212000432 (M.I. Shapovalova)

self-replicating organisms. Research was conducted on infinite one- and two-dimensional grids, and the possibility of higher dimensions was also considered. Cellular automata gained popularity in the field of computational universality and theoretical computer science, as noted in Burk's seminar collection (Burks, 1970).

In the contemporary era, the study of cellular automata has actively diversified, encompassing various disciplines. Cellular automata can be used to model various phenomena, from simple mathematical concepts to complex systems simulating biological, physical, or social behaviour. A notable example is John Conway's *Game of Life* (Conway, 1976), which drew attention from researchers in the field of automata and artificial life.

In our time, it is impossible to encompass all the research aimed at the development of the chosen theme. Therefore, it is worth noting that any review of scientific achievements will inevitably be incomplete and that this review is based on the personal experience and interests of the authors. The main focus of the work is to illuminate the extensive landscape of cellular automata research, with an emphasis on issues closely related to computer science and mathematical modeling. It is believed that such a review has not been conducted previously to this extent and will be beneficial for both beginners in the field and experts working on specific aspects of cellular automata.

Looking back in history, it is worth mentioning a series of reviews in this field that inspired the authors and served as the basis for the work (Aladyev, 1974; Smith III, 1976; Vollmar, 1977). Also, noteworthy is the review of computational theoretical aspects by Culik II et. al. (1990); and books by scholars Garzon (1995) and Chaudhuri et al. (1997), which focus on specific aspects of cellular automata research. Following Wolfram's work (Wolfram, 1986), a cellular automaton is defined as an infinite one-dimensional square cell. The next state of a specific cell is determined by its previous state and the state of neighboring cells (left and right) according to a local updating rule. A cell can exist in two states: 1 and 0, where 1 is considered a black cell, and 0 is white. Thanks to this feature, a cellular automaton can not only model biological self-replication but also be computationally universal. Simple local interactions and cell computations lead to complex behavior when interacting. Various variations of cells have been proposed to facilitate the design and modeling of complex systems.

Also worth noting is the review work by Sarkar (2000), covering four decades of cellular automata research, classifying contributions into three broad themes: classical, modern, and games. Additionally, the contemporary article by Krishna et al. (2022), focuses on the peculiarities of applying cellular automaton methods in combination with the capabilities of modern computing systems and cutting-edge technologies, including computer vision, machine learning, etc. A detailed investigation of the convergence of a one-dimensional two-state asynchronous cellular automaton with three neighbors under zero impact conditions is considered.

Understanding the breadth and diversity of the application of cellular automata in various aspects of life and fields, as well as the dynamism of the implementation of new scientific ideas, this review has decided to consolidate significant research under the following list of main topics, divided into the following sections:

- At the origins. Self-replicating automata. The universality of computation. Compromises in CA. Variants of cellular automata. CA and biological systems. Concept of fault-tolerant computation. Pattern recognition.
- CA and games.
- Cellular automata classification. Fractals in cellular automata. Dynamic properties of cellular automata. CA and complexity of calculations. Cellular automata and the number of cells.
- Image processing and CA. Patterns.
- Cryptography.
- Bioinformatics and CA.
- Cellular automata for materials modeling. Probabilistic cellular automata. Modeling recrystallization. Nucleation modeling.
- Other contemporary research using CA.
- Advantages and disadvantages of CA
- The computational cost of the CA method.
- The future application of cellular automata method.

At the origins

Initially, the cellular automaton is considered a one-dimensional array of cells (possibly two-sided infinite). Time is considered discrete, and each cell occupies one of a limited set of possible states at each step. Cells change their state at each clock tick, and the new state is entirely determined by the current state of the cell and its left and right neighbors. The function (known as the local rule) that determines this state change is the same for all cells. The automaton has no input and is, therefore, autonomous. The set of cell states at any given time is called the configuration or global state and describes the stage of the CA's evolution. At time $t = 0$, the CA is in a certain initial configuration, and it progresses deterministically under the influence of the local rule. Using the local rule for each cell of the CA leads to a change in the correspondence of all configurations in the system. This transformation is called the global mapping or global rule of the CA.

The original automaton was introduced by von Neumann (1966) as a formal model of self-replicating biological systems. A two-dimensional infinite array of homogeneous cells, where each cell is connected to four orthogonal neighbors, was initially termed a cellular space, but now the term cellular automaton is widely accepted. This contributed to the introduction of a rigorous axiomatic and deductive approach to the study of "complex" natural systems. The fundamental idea of a self-replicating automaton was presented in von Neumann's work (Neumann, von, 1951) and is a remarkable adaptation of the concept of creating a universal Turing machine (UTM).

From a mathematical perspective, a formal description of CA can be represented as a set consisting of four components:

$$CA = \langle z^d, N, A, \varphi \rangle \tag{1}$$

where:
z^d – set of d-dimensional vectors with integer coordinates (cell space);
$N = \{n_{ijk...d}\}$, $i, j, k... = 1, ..., m$ – finite set of power m of vectors with zero vector (cell neighborhood template);
A – the finite set of power k states of the cell with a dedicated state of rest $\varnothing$ state (the alphabet of the cellular automaton); In the case of, e.g., crystallization problem it is defined as $A=\{0, 1\}$.
φ – the local function of the transitions, defined in the discrete moments of time, which changes the states of the cell, which is a zero element in the template, depending on the state of the cells that form the neighborhood pattern; with $\varphi(\varnothing, \varnothing, ..., \varnothing) = \varnothing$.

Grids z^d can be of different types, differing in size and shape of cells. Each cell is a CA whose states are determined by the states of neighboring cells and its own states.

Cellular machines in general are characterized by the following properties:

- changing the values of all cells occurs simultaneously after calculating the new state of each grid cell;
- the grid is homogeneous; it is impossible to distinguish any two places on the grid over the landscape;
- interactions are local; only the surrounding cells (usually neighboring ones) can affect this cell;
- the set of states of the cell is finite.

Self-replicating automata

Consider a simplified version, disregarding encoding details formalized by von Neumann (1966), where each cell may exhibit 29 possible states. Initially, two automaton types were distinguished: the first type, automaton A, upon receiving instruction I, utilizes it for constructing an encoded automaton (or machine). The second automaton, B, simply duplicates instruction I into the control section of another automaton. Subsequently, the joint action of A and B with control automaton C was examined. A new automaton D was formed, necessitating instruction I for its operation. Following this, automaton E was created from automaton D, demonstrating self-replication. The concept of self-replication in this context originates from a single parent.

It is essential to note that a self-replicating machine must be non-trivial and capable of universal computation. If a machine can construct a set of automata, it is called a universal constructor. If this set includes the machine itself, then it is self-replicating. Vitányi (1973) describes a machine in his work that constructs an automaton from two "genetic" strings, where the offspring is not an exact copy of the parents. Amoroso & Cooper (1971) addressed the general problem of replication in cellular automata, describing CAs that, after several steps, selectively replicate their initial template.

The universality of computation

It has been revealed that a CA can systematically replicate the operations of a single-task Turing machine (TM). For analytical convenience, the assumption is made that the TM tape is infinitely long on both sides. In the simulating CA, each cell has two components: the first is for storing the symbol of the TM tape, and the second is for indicating whether the head scans the corresponding cell of the TM tape. The transition function of the TM can be easily translated into a local rule for the CA. The main idea is as follows:

- If the head is not scanning the cell or its left or right neighbor, the content of the cell remains unchanged.
- If the head scans the left cell and moves right, then on the next step, the head scans the current cell. The same applies to the opposite direction.
- If the head scans the cell, then on the next clock tick, the content of the first component of the cell is updated, and the head no longer scans the cell.

This step-by-step simulation eliminates the inherent parallelism of the CA (Smith III, 1972). Attempts have been made to isolate the power of this parallelism. Culik II et al. (1990) noted that there exists a universal CA with 14 states that can systematically replicate the actions of any other CA if its initial configuration and local rule are encoded as the initial configuration.

Compromises in CA

The initial technical challenges related to CA arose concerning various types of compromises:

– between the size of the cell (number of possible states) and the size of the neighborhood;

– between the size of the cell and the computation speed.

The idea of compromise was a direct consequence of reformulating von Neumann's original proof of self-replicating machines, which utilized 29 states per cell. Subsequent works (Banks, 1970; Codd, 1968) presented different constructions for 2-dimensional infinite CAs using the so-called von Neumann neighborhood or the 5-cell (orthogonal and the same cell). The simplest known CA with universality property, having 4 states per cell and a von Neumann neighborhood, was introduced by Banks (1970). He also provided the simplest known universal 2-dimensional CA with computational universality (3 states per cell and a von Neumann neighborhood). For a 9-cell or square unit neighborhood (also known as Moore neighborhood), 2 states per cell are sufficient, and a specific local rule called "Game of Life" demonstrated computational universality. It should be noted that reducing the size of the neighborhood or increasing the computation speed leads to an increase in the size of the state set.

Variants of cellular automata

Cellular automata are distinguished by four main aspects: the geometry of the underlying environment containing cells; the local transition rule; the states of the cells; and the cell neighborhood. Let us explore various types of CAs that can emerge due to changes in these four characteristics.

Cell states

Each cell in a CA can assume one of a finite set of possible states. Typically, there is a particular state, referred to as the quiescent state, where the local rule transitions the cell to the quiescent state if all its neighbors are also in the quiescent state. Automata where cells can have different sets of states are called polygenetic (Burks, 1970). Usually, the scenario where the sets of states for all cells are the same is considered standard. This set may have an algebraic structure. For linear CAs, the set of states is usually regarded as a field (Martin et al., 1984). CAs with state sets Zm (integers modulo m), where m is any number, are also studied (Itô et al., 1983). In the context of Very-Large-Scale Integration (VLSI), this set is considered a two-element field $\{0, 1\}$. CAs can be envisioned as a collection of finite automata. Each cell in a CA is an individual finite automaton. Although it is possible to allow each cell to assume an infinite number of states, such CAs have not been extensively studied. In the work of Litow & Dumas (1993), CAs were described where the temporal sequence of cells forms an algebraic series and thus a cell can store any integer.

Geometry

The geometry can be a d-dimensional (possibly infinite) grid. In the case of finite grids, different boundary conditions can be defined. If periodic boundary conditions are considered in a certain direction, it means that the grid wraps around in that direction. The boundary has a fixed boundary condition if the edge cells are considered neighbors in a certain predefined state that does not change during computation. If this predefined state is the quiescent state, the boundary condition is called "zero". Among the fixed boundary conditions, only the zero boundary condition has been seriously studied. However, there is work by Martin et al. (1984), providing a brief overview of other possibilities. In Bardell's work (Bardell, 1990), a case is considered where one end has periodic boundary conditions, and the other has fixed ones.

So far, static CAs have been considered – the set of nodes and the interaction scheme do not change over time. A CA can be considered static when the set of nodes does not change over time, but the interaction scheme can change (Bardell, 1990). However, dynamic CAs are characterized by changes in the set of nodes and connections (Lindenmayer, 1968).

Neighborhood

In some cases, such as group graphs, geometry itself determines the neighborhood of cells. However, if considering a d-dimensional grid, it is possible to define various types of neighborhoods (e.g., input and output neighborhoods of a cell). A cell receives input from its input neighborhood, and its state is accessible to cells in its output neighborhood. If the sizes of the input and output neighborhoods are equal, the CA is considered balanced. For balanced but non-uniform neighborhoods, interactions with a uniform neighborhood were studied by Jump & Kirtane (1974). A variant of CA where the local rule depends on the sum of states of neighboring cells is called "totalistic" (Culik II et al., 1990).

Yamada & Amoroso (1969) considered tessellation automata, with an input sequence distributed among all cells. This construction can be imagined as if each cell has a finite set of local rules, and the input is used to select a specific local rule. The completeness problem for this class of automata was discussed by Mahajan (1992).

Another class of CAs includes iterative automata, where only a specific cell is given input. Variants using such automata in studying language recognition and various compromise results for this class have been explored (Kosaraju, 1975; Seiferas, 1974). Smith III (1976) also noted that iterative automata may be less efficient compared to regular CAs.

Works by Chang et al. (1986) and Ibarra et al. (1984) featured the use of one-dimensional CAs with one-sided communication, where each cell depends only on itself and its left neighbor. It was emphasized that such limited one-sided information flow may seem restrictive for the power of the automaton, but results are provided indicating universality in the case of one-dimensional, one-sided reversible CAs.

Linguistic properties of one-sided CAs, including their ability to recognize PSPACE-complete languages and languages accepting an alternative computational machine time, were addressed in a series of papers (Kari et al. 1997; Róka, 1994). The connection of one-sided iterative automata with complexity theory was discussed. Additionally, the connection with systolic tissue automata, applied in designing systolic systems and algorithms, was indicated.

CA and biological systems

Let us now turn to the use of cellular automata in modeling biological systems. It should be emphasized that cellular automata were proposed by von Neumann (1966) for the formal analysis of "complex" natural systems. Subsequent works in this direction included structures known as dynamic CA, used for modeling biological systems where cells can appear or disappear (Kari et al., 1997). Modeling the growth of filamentous organisms, including branching structures, was considered. The concept of *L*-systems and their application in modeling plant life was presented by Prusinkiewicz & Lindenmayer (1990).

Implementations of self-replication and artificial life constructions in the context of cellular automata were addressed in the works of Langton (1984) and Pesavento (1995). The emergence of self-replicating systems was investigated in a model where CA is used to simulate the universe. Each cell has two parts: the first stores the cell's state, and the second indicates the strength of connections with neighbors. Randomness operators are used to process states based on connections. A parameter determining the expected time for the occurrence of self-replicating systems was studied by Holland (1976).

Concept of fault-tolerant computation

Incorrect cell operation can be conceptualized as a failure of components. A model of errors was introduced using the *k*-separated misoperation model, and it demonstrates how to construct a CA (Nishio & Kobuchi 1975), that can accurately simulate a reliable CA under the conditions of this error model. The significance of this approach is emphasized from the perspective of ergodic theory, with important implications highlighted for statistical physics, as evidenced by Gács's results (Gács, 1986). He illustrated the construction of a 1-dimensional CA capable of reliably performing arbitrarily large computations with an error probability assigned to each cell. The importance of this achievement is underscored in the context of ergodic theory, with significant implications for statistical physics.

Pattern recognition

It has been noted that a finite cellular automaton can be regarded as a language acceptor, where the initial configuration is considered an input string, and acceptance or rejection is determined by the state of a specific cell. For 2D CAs, the challenge lies in pattern recognition, where the decision-making cell may be selected, for instance, in the northwest for a rectangular grid or on the eastern edge cell in a northern row for a general 2D composition (Dyer, 1980). Previous research (Smith III, 1972) established that linear, context-free, and context-sensitive languages can be accepted by CAs (including one-sided CAs) in real-time. The language classes of CAs are defined by limiting and enhancing their computational power, taking into account one-sided communication, real-time and linear-time operation, as well as non-determinism. Examples of such language classes were provided along with the results of their study (Mahajan, 1992).

A series of works was dedicated to investigating issues of invertibility, surjectivity, and the "Garden of Eden". Conditions for invertibility were explored for CAs (Richardson, 1972), defined by a rule *r*, considered invertible if there exists another rule *r*21 (inverse rule) that brings the CA back. The problem of the surjectivity of the global mapping of CAs and the configuration of the "Garden of Eden" (Myhill, 1963), which cannot be achieved through CA evolution, was also discussed. Results regarding invertibility and surjectivity were presented for different types of CAs, including linear, two-dimensional, and group-based. The question of invertibility was deemed a significant one in physics, as it can be utilized to model microscopic reversible dynamic systems (Harao & Noguchi, 1978).

CA and games

The synchronization problem in cellular automata can be viewed as a game. Culik II & Dube (1991) set the task of synchronizing the actions of *n* soldiers (with one general) in a row so that they simultaneously execute the command to fire. The problem was modeled using CA, and the solution involved creating cells and a local rule that ensures the simultaneous and first-time execution of the fire command. Various approaches to solving this problem have been noted, including the use of signals propagating at different speeds through the array and the minimal state to address the task. The problem was also examined in the context of higher dimensions and other modifications (Connelly, 1986), with the solution referred to as the Fire Brigade Theorem. Research results have been applied in language studies and pattern recognition in CAs (Sarkar & Barua, 1998).

The pivotal event during the mid-1980s in CA research was the experimental investigation of growth patterns conducted by Wolfram (1983). A paradigmatic shift in CA research was explored, considering their nature as models of complex systems that can emerge from very simple rules. The phenomenon of self-organization in CA evolution was studied, where the system evolves from chaotic initial configurations to states of lower entropy. This contradicts the second law of thermodynamics, and its microscopic irreversibility induces self-organizing behavior.

In particular, binary CAs with three-dimensional neighborhoods and rules 90 and 150 were studied, where local and global statistical parameters are defined to examine growth patterns. Entropy and its reduction for irreversible and second-order reversible CAs were also investigated. Using formal language theory, it was demonstrated that the set of configurations that can arise after *t* steps form a regular language, although some CAs can generate irregular languages within the limit (Green, 1987).

CA classification

The classification of cellular automata can be examined based on their behavior, which emerged against the backdrop of Wolfram's works. The initial classification, proposed by Wolfram himself, relies on measures of entropy and categorizes CAs into four classes depending on their evolutionary outcomes. Subsequently, efforts were directed toward formalizing intuitive classifications, and Culik II & Yu (1988) proposed a more precise classification as follows:

- Rule *r* belongs to class one if and only if every finite configuration, i.e., configurations where only a finite number of cells are in unsettled states, evolves to a stable configuration within a finite number of steps.
- Rule *r* falls into class two if every finite configuration evolves to a periodic configuration within a finite number of steps.
- Rule *r* belongs to class three if one can determine whether a configuration is encountered on the orbit of another.
- The fourth class encompasses all local rules.

Research indicates that determining the membership of rule *r* in the first and second classes is a task of level Π_1^0 complexity. Similarly, the third class is a Σ_1^0-level task. In Sutner's work (Sutner, 1989), they demonstrated that the first and second classes are Π_2^0-complete, and the third class is Σ_3^0-complete. The arguments were based on encoding instantaneous descriptions of Turing machines with natural numbers and modeling using CAs. It is crucial to note that the above classification only considers finite configurations. Infinite configurations cannot be described as finite and, therefore, cannot be resolved using traditional computability theory. A classification of periodic boundary conditions in cellular automata was also proposed. Sutner (1990) introduced configurations that can be regarded as spatially periodic configurations of an infinite CA. Research by Braga et al. (1995) provided a classification of CAs based on pattern growth dependent on the truth table of the local rule. This allowed the determination of an effective hierarchy of CA rules, addressing the uncertainty issues discussed earlier. The technique of detecting certain shift-directed dynamics in evolution was significant through the study of the truth table of the local rule.

Analyzing 2D CAs, Packard & Wolfram (1985), determined that it is possible to classify them based on the same principles as 1D CAs. Such behavior indicates a similarity in the global behavior of both types. However, differences have been identified between them in other properties, such as the presence of recursive configurations in 1D CAs compared to their absence in 2D CAs.

Recent research highlights the importance and relevance of the CA classification task. The identified challenges underscore the need for further exploration and systematization of this knowledge domain. In particular, the work of Vispoel et al. (2022) distinguishes a dichotomy in CA research and reveals two main directions: theoretical, focused on topological dynamics and computation theory, and experimental, centered on statistical properties of simulated patterns. Terry-Jack & O'Keefe (2023) introduce a 0–1 test for the automatic classification of elementary cellular automata, leading to the discovery of chaos in the majority of rules and revealing unexpected properties in complex rules. Meanwhile, Salo et al. (2022) concentrated on the qualitative properties of two-dimensional "freezing" cellular automata, establishing monotonicity constraints on their asymptotic dynamics and presenting novel results.

Particularly noteworthy are the works of Wolnik et al. (2023), exploring non-periodic elementary cellular automata on an infinite lattice that preserves numbers. Such CAs operate in a one-dimensional environment, where individual cells use different Wolfram rules to change their states. The research results allow for a comprehensive description of these cellular automata capable of preserving numbers on an infinite lattice, distinguishing them from previously known properties on finite grids.

Fractals in cellular automata

In exploring another direction of cellular automata research, attention was focused on constraint sets and fractal properties aimed at studying the spatiotemporal patterns of CAs. Early contributions in this direction were made by Wilson (1984) and Wolfram (1983), but the concept of the constraint set of configurations obtained through CA evolution was introduced by Podkolzin (Culik II et al., 1990). A fractal is a geometric figure with self-similarity and chaos characteristics. The authors provided insights into the fractal dimension of spatiotemporal patterns of CAs, using two approaches proposed by Wolfram. One utilizes the parameter $T(n)$, measuring the density of triangles with side length n, and the other employs geometric construction for scaling spatiotemporal configurations and considering the set of all constraint points.

Theoretical investigations of constraint sets were conducted by Wilson (1984), where the study object is the sequence ω, $F\omega$, $F^2\omega$, ..., $F^p\omega$, ..., where ω is the configuration of an n-dimensional CA, and F is the global rule of another CA. For linear CAs, it is shown that the constraint set is a compact subset of Euclidean space and may have a fractal dimension. However, for linear CAs, the constraint set can generally be a fractal. Conditions for constraints on configurations and their undecidability were also examined. Work on CA dynamics involves defining a state transition diagram, indicating that spatiotemporal configurations of CAs evolve in fractal sets with a complex structure, investigated using various methods such as geometric invariants and formal language methods.

In some works (Aruldoss & Pricilla, 2014; Bruno, 1994; Gütschow et al., 2010; Ni, 2003), properties of fractals, such as self-similarity and chaos, have been discussed, along with their determination using fractal dimension. The use of CAs for generating fractals, including linear totalistic CAs and lattice automata, was explored and methods for determining fractal dimensions, such as self-similarity and box-counting dimension, were described. Finally, issues in studying the fractal behavior of CAs were considered, together with a presentation of methods for addressing these issues.

Dynamic properties of cellular automata

When discussing the dynamics of cellular automata, various aspects of studying the dynamic properties of cellular automata have been considered. One approach involves considering the automaton as a computational device acting on bidirectional sequences and as a continuous function in a compact metric space. This leads to the examination of symbolic dynamics on bidirectional sequences, introducing the concepts of "subsystems" and "Sophic systems" (Culik II & Yu, 1988). It was established that each Sophic system is an ωω-regular set, and for the global mapping of CA, $Gi(SZ)$ holds for each $i >= 0$.

Additionally, it is noteworthy to consider other characteristics of CAs that researchers have actively investigated: topological properties of transitivity, sensitivity to initial conditions, attractors, expansiveness, topological entropy, and Lyapunov exponents. Numerous interesting results have been obtained, Hurd et al. (1992) discovered that the topological entropy of cellular automata is incomputable. For linear and positively expansive cellular automata, this can be computed, as demonstrated in the work of D'amico et al. (2003). Kurka (1997) examined attractors of CA, and linear cellular automata were investigated in the study by Manzini & Margara (1999). The relationship between Lyapunov exponents and expansiveness and sensitivity was explored in the work of Finelli et al. (1998). A classification of CAs into five non-intersecting classes based on the structure of their attractors was performed by Kurka (1997).

CA and complexity of calculations

Another crucial aspect of working with cellular automata is the concept of computational complexity. When addressing the task of determining the minimum number of steps required to perform specific computations in a CA, the authors devised a certain approach (Wolfram, 1984). It involves constructing a graph to represent the temporal steps and the set of constrained configurations. Using this graph, one can view the CA as a state transition graph of a finite automaton. As a consequence of Wolfram's result, the existence of predecessors for 1-dimensional CAs is decidable (i.e., given a configuration *X*, one must check if there exists a configuration *Y* that evolves to *X* in a one-time step). Subsequently, the computational complexity of CAs was investigated, particularly in the search for *NP*-complete problems. Some of the initial findings appeared in Green's work (Green, 1987), where it was demonstrated for a specific CA that determining subsets, their recurrent occurrence, and the possibility of generating a certain sequence defined by states are *NP*-complete problems. The research results indicate that the computational complexity of CAs can significantly vary depending on specific conditions such as dimension and type of the CA.

CA and the number of cells

Cellular automata are confined by the number of cells. For such CA, dynamic properties are entirely determined by the State Transition Diagram (STD), which is a directed graph. In this graph, nodes represent CA configurations, and edges signify the transition from node *i* to node *j* in one time step. As a constrained CA is a deterministic machine, the STD consists of components, each having a unique cycle and trees of height >= 0 emanating from the cyclical vertices. The cycles reflect the system's stable behavior and are sometimes referred to as attractors, while branches in the tree depict the initial transient behavior. Crucial questions arise regarding the dynamic parameters of the system: the number of cycles, cycle lengths, tree heights, branching degree of each node, etc. For some confined CA, important questions like reversibility and maximum cycle length have been explored in the works of Harao & Noguchi (1978).

In the case of linearly constrained CA, more information can be obtained using algebraic methods. The STD in this scenario reveals a more homogeneous behavior, where trees emanating from any cyclical vertex are isomorphic to the tree emanating from the zero configuration. The degrees of all nodes are identical and equal to the dimension of the linear map's kernel, etc. For 1D periodic constrained CA, many results exist, addressing the reachability of configurations in one step from configurations composed solely of units for 2D CA (Aso & Honda, 1985). Algebraic methods, such as polynomials and matrix representation, are employed to analyze the behavior of linear CA. The results of these studies indicate that CA analysis can be crucial for understanding their dynamic and algebraic behavior, as well as for applications in Very Large Scale Integration (VLSI) technologies (Litow & Dumas, 1993).

The application of constrained CA in VLSI technologies involves generating pseudorandom sequences for Built-In Self-Tests (BIST) (Chaudhuri et al., 1997). The configurations of CA are considered as a random sequence, proving to be a successful method in VLSI. Additionally, CAs have been utilized in areas such as error-correcting coding, finite state machine testing, secret-key cryptosystems, and associative memory design.

In VLSI, the most prevalent use is of 1D binary cellular automata, although the utilization of 2D structures has also been noted. Linear or affine maps are primarily employed, as nonlinear ones are too complex to analyze. The condition of a zero edge is crucial, as periodic conditions require "long-range connectivity" between edge cells (Bardell, 1990). Typically, the CA structure is hybrid, where each cell has its own rule. Regarding theoretical issues associated with hybrid 1D CAs, designing 90/150 CAs with a zero-edge condition based on a specified irreducible or primitive polynomial, which serves as the characteristic polynomial for the CA, is a significant task in VLSI (Niemi, 1997; Tezuka & Fushimi, 1994).

Image processing and CA

The extensive scope of research explores the application of cellular automata in image processing. Investigations in this direction have been ongoing for the past seventy years, starting with significant contributions from scientists such as von Neumann (1951), Wolfram (1983), and Conway (1976). The transition from developing hardware specifically designed for CAs to their utilization in a wide range of computer vision tasks, including image analysis (Nayak et al., 2014), has been highlighted. In recent years, CAs have been successfully applied in image processing, particularly in parallel algorithms, which constitute a crucial direction in the modern world (Rosin, 2005). Linear CAs dominate image processing tasks, although hybrid and nonlinear CAs still pose a challenge for researchers (Dioșan et al., 2017). It is noted that in image processing, a primary role is attributed to 2D CAs, where image pixels are represented as cells that synchronously update their states at each step. Various CA rules are employed for different image processing tasks, including noise filtering, edge detection, image compression, and other functions (Popovici & Popovici, 2002). The successful use of CAs in various image processing tasks, such as noise reduction, contour detection, compression, segmentation, and more, establishes them as significant tools in many modern applications, including satellite television, computer tomography, and other fields of science and technology (Dioșan et al., 2017).

Patterns

For simple pattern recognition, a special type of cellular automata with multiple attractors (GMACA) has been introduced. This is a promising pattern classifier that utilizes a simple local network of elementary cellular automata (ECA). In Wolfram's work, the attractor basin was defined as representing an inverted tree-like graph. To organize the rules of CA, GMACA employs the reverse design method and a genetic algorithm (GA), leading to a significant drawback in computational complexity and recognition performance (Wongthanavasu & Ponkaew, 2013). Due to these drawbacks, it was decided to present a classifier based on binary CAs, known as the two-class classifier GMACA with an artificial point (2C2-GMACA), which proved to be 7-14 times more efficient.

Combining analysis and contextual information with artificial neural networks. As demonstrated by Brady et al. (1989), an iterative framework for a neural network that allows the incorporation of basic resonance into the network's topology was proposed. The use of artificial neural networks, such as the Hopfield Network, enables effective problem-solving in pattern recognition. This approach also considers correlation and possible noise, ensuring the integration of object structure information in a parallel iterative algorithm.

Maji & Chaudhuri (2005) explored the utilization of fuzzy cellular automata (FCA) for image classification. FCA operates on binary strings using Boolean logic and, using an algebraic metric formula, FCA was developed for analysis and synthesis. Experiments confirmed the scalability of the FCA classifier for processing large datasets. These cellular automata have alsoi been examined in theory and practice, configuring themselves through rules with *OR* and *NOR* logic.

A specialized class of one-dimensional cellular automata, known as linear additive CAs, has attracted the attention of many researchers (Das et al., 2009). They have been recognized and successfully applied in various domains such as VLSI design, cryptography, and pattern recognition. For efficient pattern recognition, a multi-cycle attractor with a single length is employed, which remains relevant to the present time. To enhance the performance of the pattern recognizer, a corresponding scheme has been developed to consider a single-length attractor and avoid attractors of different lengths (Das et al., 2009).

Maji et al. (2003) devised a classifier based on a specific type of Sparse Network CA. This classifier can be applied in various areas, including data mining, image compression, and fault diagnostics. Defining objects into different classes in a database is a key aspect of data or image classification. The primary requirements for classifier development today include high throughput and low storage demands. Classical methods such as Bayesian classification and neural networks prove to be overly complex. Therefore, a classifier based on a sparse network is proposed, reducing the algorithm complexity based on cellular automata from $O(n^3)$ to $O(n)$.

Cryptography

Cryptography is a fundamental technique for ensuring the security of data storage and transmission in global electronic communication networks. Various aspects of cellular automata (CA) applications in cryptography have been widely discussed over the years (Guan, 1987; Gutowitz, 1993; Sadiq & Kumar, 2015; Tomassini & Perrenoud, 2001), particularly in one-dimensional and two-dimensional non-uniform CAs for generating pseudorandom bit sequences. The primary categories of cryptographic algorithms, such as symmetric, asymmetric, and authentication algorithms, have been highlighted, along with their main tasks and operational principles. It was noted that symmetric algorithms use the same key for encryption and decryption processes, while asymmetric algorithms employ different keys for these processes. The importance of authentication algorithms for verifying the sender's identity has also been emphasized (Sadiq & Kumar, 2015).

A detailed description of the use of CAs in various areas of cryptography, such as ensuring the confidentiality of communications, developing block and stream cyphers, and applying invertible CAs to construct efficient cryptographic systems (Bouchkaren & Lazaar, 2014). It was underscored that the use of CAs in cryptography can provide efficient cryptographic systems capable of working in parallel and at high speed, making them competitive compared to existing encryption systems. Research on linear cellular automata has shown the possibility of creating error correction codes (both single-bit and multi-bit) and protection against attacks (Maiti et al., 2021).

Note that some research in the field of cryptography has focused on synchronous cellular automata, and in this context, asynchronous CAs were first considered by Sethi & Das (2016), which update completely asynchronously. The authors described a proposal for a new symmetric cryptosystem based on asynchronous CAs, utilizing the property of feedback transactions for encryption and decryption. A comparative analysis with existing cryptographic systems was conducted, emphasizing the competitiveness of the proposed system. Using simple cellular automata, Fúster-Sabater & Caballero-Gil (2009) successfully linearized nonlinear sequence generators.

One of the peculiarities of cryptographic applications is ensuring integrity and identity authentication. The behavior of nonlinear and non-group cellular automata with periodic boundary conditions was discussed by Jeon (2010). This allowed the identification of CAs with a sufficient number of unique states and the extraction of specific states from the automaton based on properties such as Hamming distance, frequency, growth, and decay. Isa et al. (2014) utilize I/O automata, a technology that ensures the integrity protection of a secret key, enabling the efficient implementation of the Message Authentication Code (MAC) cryptographic protocol for key exchange, especially in the context of wireless sensor networks. Roy et al. (2016) proposed the use of cellular automaton rules in wireless sensor networks. The application of a symmetric block cipher method, based on rules of non-complementary automata and a hybrid vector of rules, contributes to efficient data encryption and decryption. In the work of Dennunzio et al. (2024), an effective algorithm for determining chaos in linear cellular automata is presented. This algorithm is relevant for cryptographic applications and is used to create chaotic systems and enhance existing methods.

Analysis of works by Mohamed (2014); Bhardwaj & Bhagat (2018); and Khedmati et al. (2020), indicate a wide range of methods and approaches to cryptography and information protection in digital images. The use of 2D automata, Arnold transformations, reversible cellular automata, and hybrid chaotic maps demonstrates the diversity and effectiveness of cryptographic methods in the context of image security. Experimental investigations confirm the high level of security and performance of the applied algorithms, making them potentially valuable for real-time applications.

Bioinformatics and CA

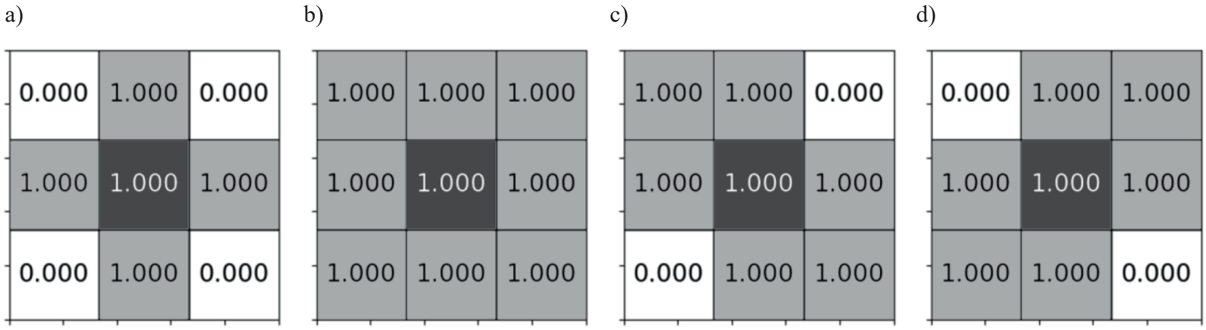
Bioinformatics, encompassing a broad spectrum of tasks ranging from data storage and recovery to the identification and analysis of features within them, also highlights the distinctive role of CA in the field. Sree et al. (2014) introduced the concept of logical connection among various issues in this domain and attempt to develop a unified framework for their resolution, including protein encoding, promoter prediction, and protein structure analysis. The fundamental characteristics of CA, its rules, and extensions such as the Artificial Immune System-based classifier and Multiple Attractor Cellular Automata (AIS-MACAs) are examined. The conclusion emphasizes the potential effectiveness of using CA in bioinformatics, particularly for addressing protein encoding, promoter prediction, and protein structure tasks:

- **Protein encoding:** the primary task involves identifying regions that contain instructions for creating proteins within the cell. Sree & Babu (2008) introduce the FMACA algorithm based on the Undirected Fuzzy Multiple Attractor Cellular Automaton (Fuzzy MACA). The algorithm employs a pattern classifier to determine the coding region of the DNA sequence. The K-Means algorithm is utilized to enhance the classifier. Experimental results confirm the efficiency and scalability of the proposed classifier.
- **Promoter region prediction:** determining promoter regions is a crucial step in understanding human genes. Sree & Babu (2010) present a new text clustering algorithm based on CA to indicate these promoter regions in genomic DNA. Experimental results confirm the applicability of the algorithm for identifying promoter regions, showing a 12% increase in accuracy for shorter DNA sequences.
- **Protein structure prediction:** the research is dedicated to predicting the protein structure from amino acid sequences (Sree et al., 2013). It notes that most existing approaches are sequential and developed for classifying input data into four different classes. The author develops the MACAs classifier, which is applied to each of the ten classes. Experiments show that PSMACAs provide the highest accuracy ranging from 77% to 88.7%, depending on the dataset.

Cellular automata for materials modeling

CA is widely employed for modeling the structure of polycrystalline, composite, and mesoporous materials. However, this field has its peculiarities. The use of symmetric neighborhoods in crystallization tasks is conventional. Depicts typical boundaries for a two-dimensional cellular automaton presented in Figure 1 (Vodka, 2019; Wang W., 2003). Symmetric boundaries, including von Neumann & Moore, are presented here (Fig. 1a, b), as well as asymmetric ones like the left and right neighbors (Fig. 1c, d). The utilization of such boundaries can be applied to construct microstructures with specific grain orientations ±45°.

Similarly, this can be extended to multidimensional cases. Figure 2 illustrates symmetric neighborhoods: von Neumann (a), radial (b), and Moore (c).


a)

0.000	1.000	0.000
1.000	1.000	1.000
0.000	1.000	0.000

b)

1.000	1.000	1.000
1.000	1.000	1.000
1.000	1.000	1.000

c)

1.000	1.000	0.000
1.000	1.000	1.000
0.000	1.000	1.000

d)

0.000	1.000	1.000
1.000	1.000	1.000
1.000	1.000	0.000

Fig. 1. Cell neighbors: a) von Neumann; b) Moore; c) left neighbour; d) right neighbour

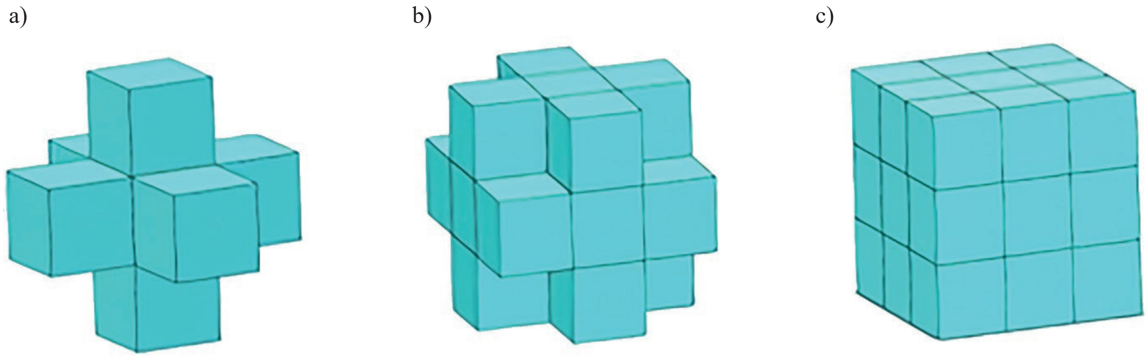
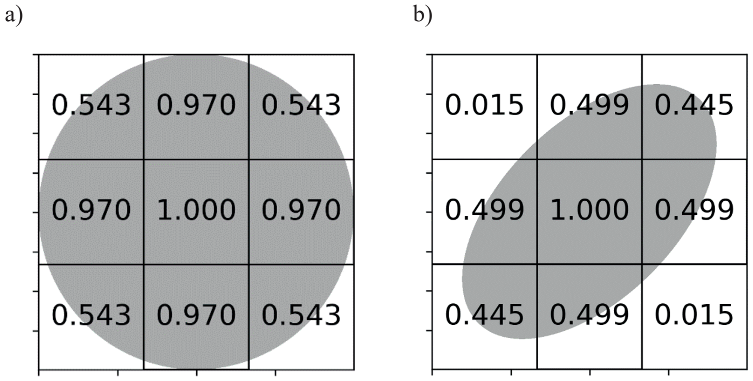


Fig. 2. Neighborhoods of regular cells in a 3D: a) 3D von Neumann, b) 3D radial; c) 3D Moore (Deng et al., 2022)

Probabilistic cellular automata

Over time, it became evident that well-known neighborhoods could lead to the artificial distortion of grain geometry and thus the expansion of cellular automata usage was necessary. The next step in this direction involved the utilization of probabilistic CA (Hallberg et al., 2010; Popova et al., 2015; Raabe & Becker, 2000; Vodka, 2020). The cited works employed a similar approach, where the occurrence of neighbors is determined by a certain probability. This probability can be predefined or calculated according to specific patterns. According to Vodka (2019), the use of probabilities for neighbor occurrence is prohibited. This allows for controlling the direction of crystallization (Fig. 3). Corresponding cells indicate probabilities of neighbor occurrence, determined by the projection of a circle and ellipse onto the field. This approach can be extended to 3D or higher dimensions.

A comparison of microstructures obtained using different types of neighborhoods is shown in Figures 4 and 5.


a)

0.543	0.970	0.543
0.970	1.000	0.970
0.543	0.970	0.543

b)

0.015	0.499	0.445
0.499	1.000	0.499
0.445	0.499	0.015

Fig. 3. Probability of the neighborhood: a) circle; b) ellipse

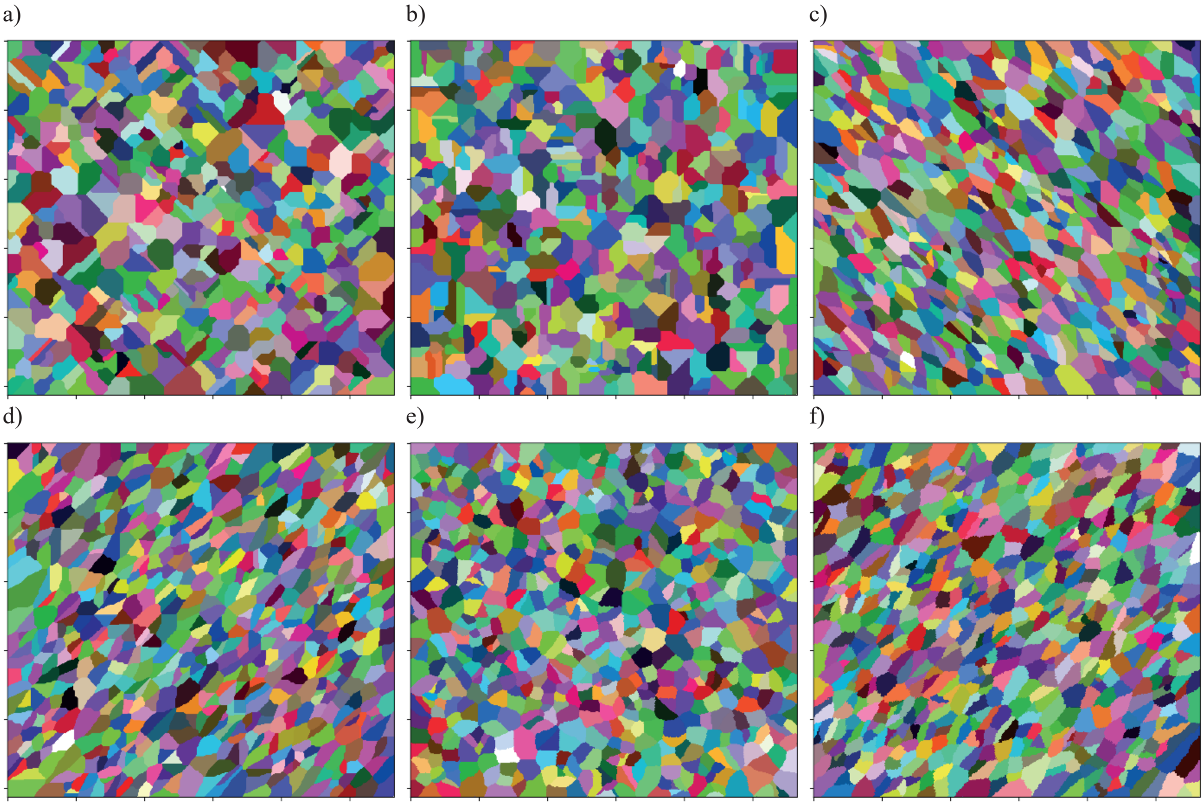


Fig. 4. Results of microstructure generation by the method of cellular automata with different neighbors: a) Moore; b) von Neumann; c) left neighbor; g) right neighbor; d) probabilistic method (circle); e) probabilistic method (ellipse)

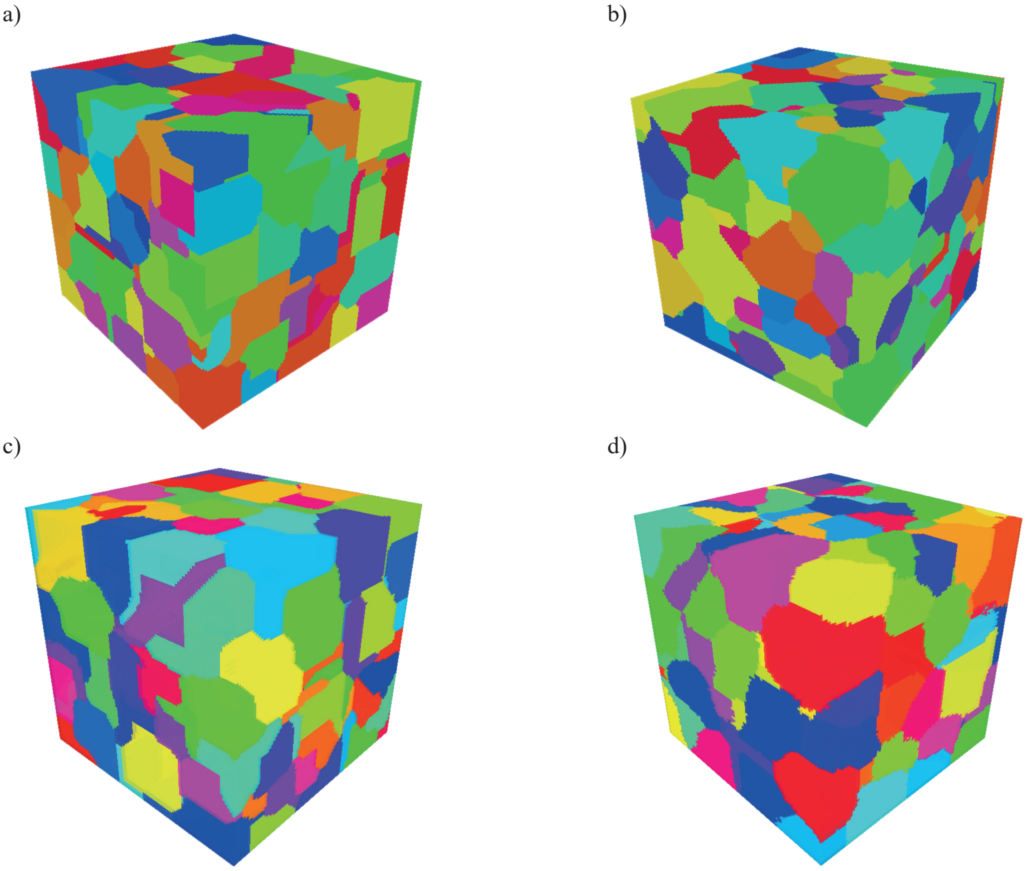


Fig. 5. Results of microstructure generation by the 3D method of cellular automata with different neighbors: a) Moore; b) von Neumann; c) probabilistic circle; d) probabilistic ellipse

Modeling recrystallization

Research on modeling recrystallization using cellular automata began alongside the development of computational techniques. Among the initial works in this direction, notable contributions include the studies by Hesselbarth & Göbel (1991), Goetz & Seetharaman (1998), Raabe & Becker (2000), Ding & Guo (2002). These works employed classical approaches to model the recrystallization process. To address the recrystallization task, solutions for heat conduction, crystallization processes, and the stress state are required. Therefore, the next step involves combining cellular automata with modeling using the finite element method and crystal plasticity models (Cui et al., 2023; Li et al., 2016; Zhao et al., 2013). This significantly enhances modeling accuracy, allowing for the assessment of a grain size's impact on stress states deformation patterns, and/or other mechanical characteristics. A series of works dedicated to modeling recrystallization in specific materials, considering their peculiarities, should be emphasized:

- 42CrMo steel (Chen M.-S. et al., 2017);
- HY-100 steel (Qian & Guo, 2004);
- medium carbon Cr-Ni-Mo alloyed steel (Zhang et al., 2016);
- Al-Cu alloy (Lee et al., 2022);
- titanium alloys (Li et al., 2016);
- magnesium alloy (Wang L. et al., 2018).

A detailed overview of recent works in the direction of recrystallization is provided in the study by Madej & Sitko (2022). Special attention should be paid to works addressing the development of software for modeling recrystallization. In the study by Baran et al. (2024), the influence of random number generators on the process of modeling recrystallization is investigated. Additionally, significant emphasis has been placed on developing and exploring algorithms for modeling the crystallization process on GPU (Sabau et al., 2023).

Dendritic structures

Modeling dendritic structures is a distinct direction in material modeling, involving the creation of CA for generating dendritic formations. A typical depiction of a dendritic structure is illustrated in Figure 6.

The solution to the problem of modeling dendritic structures is associated with the diffusion Equation (2), where $C_{l,s}$ represents the concentration of the liquid and solid phases, $D_{l,s}$ are diffusion coefficients of liquid and solid phases, ∇ – nabla operator, indexes: s – for solid, l – for liquid (Reuther & Rettenmayr, 2014).

$$\frac{dC_{l,s}}{dt} = \vec{\nabla}(D_{l,s} \cdot \vec{\nabla} C_{l,s}) \tag{2}$$

The velocity (υ) solid-liquid interface is determined by Equation (3):

$$\upsilon = \frac{1}{C_l^\Gamma - C_s^\Gamma}\left(D_s \frac{dC_s}{d\vec{n}}\bigg|_s - D_l \frac{dC_l}{d\vec{n}}\bigg|_l \right) \tag{3}$$

The cellular automaton method is employed as a solution technique for these equations. In the current modeling stage, the predominant approach involves 3D simulations (Liu et al., 2021; Zhao et al., 2014, 2015), and the finite volume method has been utilized to enhance modeling accuracy (Dobravec et al., 2017).

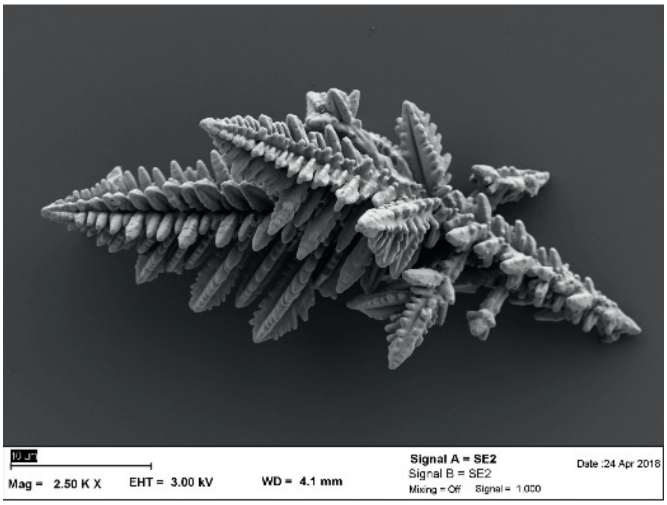


Fig. 6. A pure copper crystal with a dendritic structure (Wikipedia, n.d.)

Nucleation modeling

The majority of the reviewed studies employ the classical model (Lewis et al., 2015) (4), which relies on the Gibbs energy (ΔG), undercooling temperature (T), and the Boltzmann constant (k).

Introducing models in discrete space and time of cellular automata can be challenging. To evaluate the influence of crystallization parameters on the geometric characteristics of the formed microstructures, generalized models (5)–(9) have been proposed (Vodka, 2020). These models assume that the number of crystallization centers N_{gr} for all models has the same value at N_p iteration of the algorithm. The nucleation rate is defined as a derivative of the number of grains $N(n)$ function, where n is the iteration number of the algorithm. The visualization of the proposed models is shown in Figures 7–13.

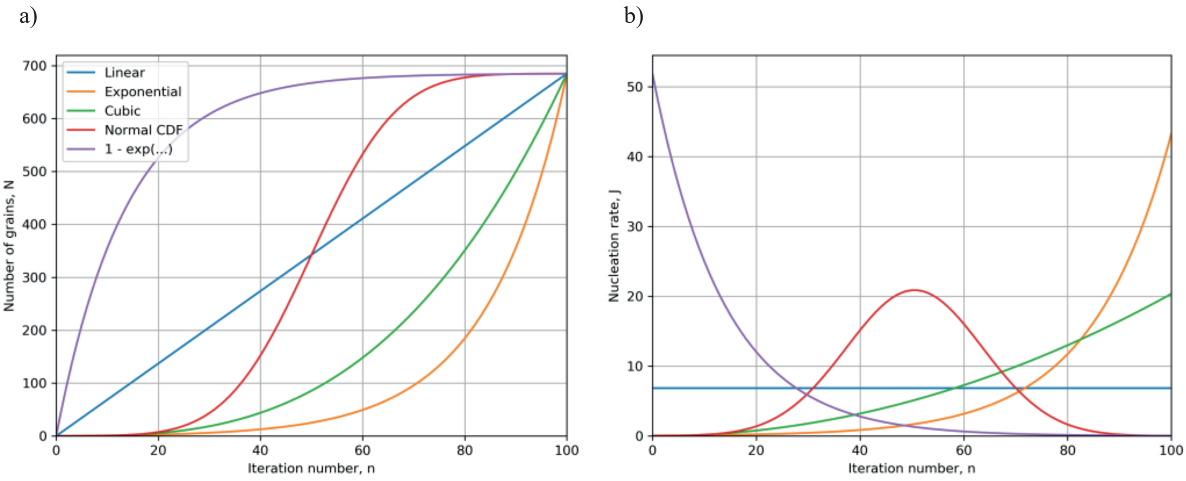


Fig. 7. Number of grains (a) and nucleation rate (b)

$$N(n) = N_{gr} \qquad (4)$$
$$J = N_{gr}\delta(n)$$

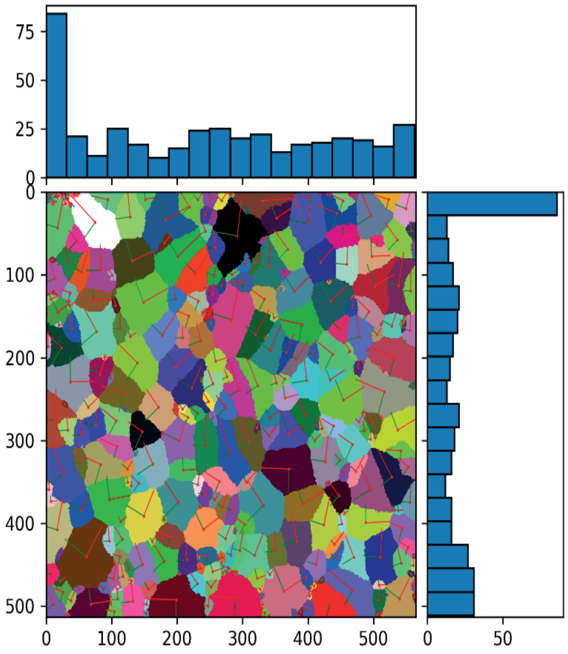


Fig. 8. Microstructures generated by the cellular automata method (without nucleation model)

$$\begin{gathered} N(n) = \frac{N_{gr}}{N_p^{\,3}} n^3 \\ J(n) = \frac{3N_{gr}}{N_p^{\,3}} n^2 \end{gathered} \qquad (5)$$

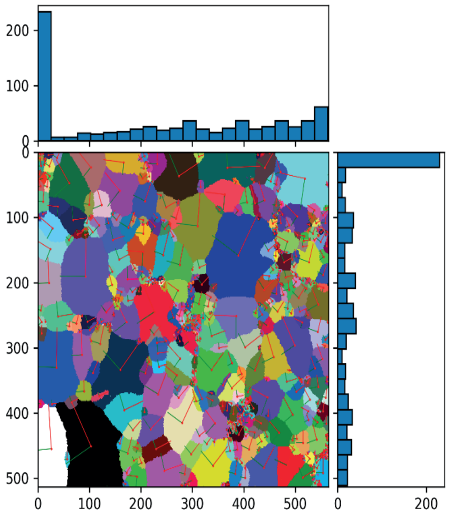


Fig. 9. Microstructures generated by the cellular automata method

$$\begin{gathered} N(n) = \frac{N_{gr}}{N_p} n \\ J(n) = \frac{N_{gr}}{N_p} \end{gathered} \qquad (6)$$

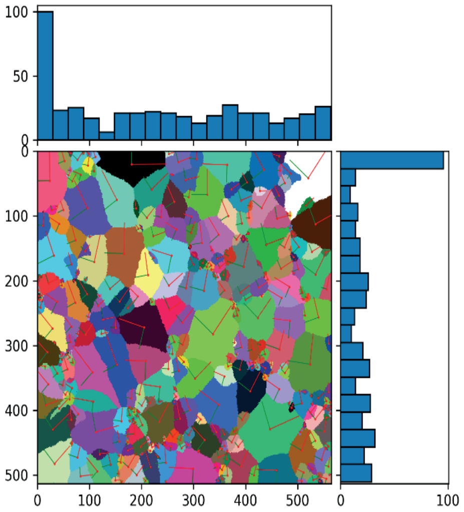


Fig. 10. Microstructures generated by the cellular automata method

$$\begin{gathered} N(n) = \exp\left(\frac{\ln(N_{gr})}{N_p} n\right) \\ J(n) = \frac{\ln(N_{gr})}{N_p} \exp\left(\frac{\ln(N_{gr})}{N_p} n\right) \end{gathered} \qquad (7)$$

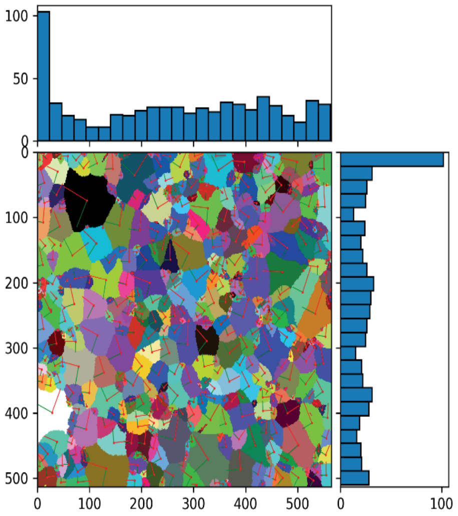


Fig. 11. Microstructures generated by the cellular automata method

$$N(n)=\frac{1}{2}\left[1+\operatorname{erf}\left(\frac{n-N_p/2}{\sqrt{N_g}}\right)\right]$$
$$J(n)=\frac{1}{\sqrt{N_{gr}\pi}}\exp\left(\frac{N_p/2-n}{\sqrt{N_{gr}}}\right) \tag{8}$$

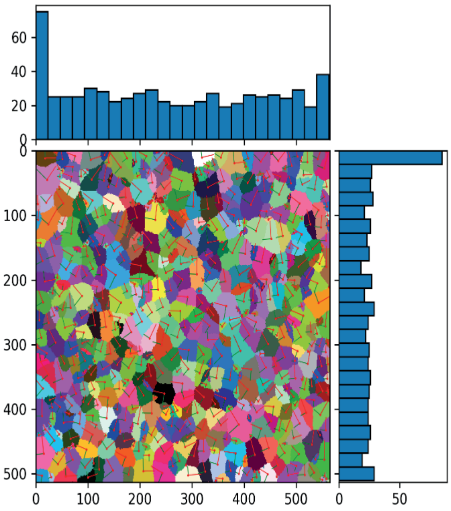


Fig. 12. Microstructures generated by the cellular automata method

$$N(n)=N_{gr}\left[1-\exp\left(-\frac{N_p}{2N_{gr}}n\right)\right]$$
$$J(n)=\frac{N_p}{2}\exp\left(-\frac{N_p}{2N_{gr}}n\right) \tag{9}$$

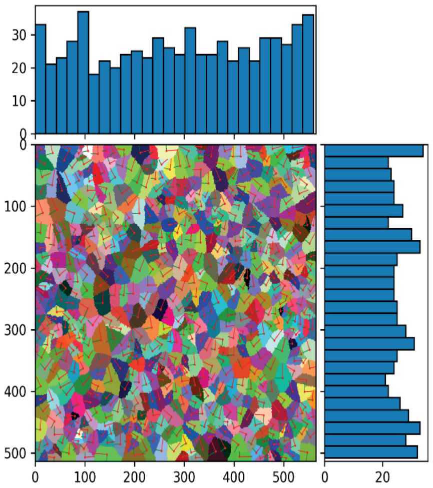


Fig. 13. Microstructures generated by the cellular automata method

Other contemporary research using CA

Cellular automata are mathematical models composed of a grid of cells in different states. The application of these automata spans various domains, from biological modeling to manufacturing technologies. In biology, they are used to model evolutionary (Aburas et al., 2016) and genetic processes, revealing interactions between different species and genetic elements. In computer graphics, cellular automata allow for the creation of complex graphical effects, from simulating fire to texture propagation. In Efstathiou et al. (2023), the authors present a stochastic model that combines a cellular automaton for forest fires with random walks for flame particles and hot gases. Mastorakos et al. (2023) combined the probabilistic-density function method and cellular automaton is proposed for simulating the spread of forest fires in areas with heterogeneous composition. The method considers turbulent convection, diffusion of hot gases and flame particles, wind, and interaction with neighboring fire fronts.

In the field of modeling and prediction, cellular automata are used in economics to analyze market interactions and in ecology to study the impact of environmental changes on species distribution and ecosystems (Ghosh et al., 2017; Mi et al., 2023). For example, Sapino et al. (2023), address the issue of water markets and their potential to ensure efficient resource distribution in conditions of scarcity. An innovative CA model for predicting the transport of floating plastic waste in the marine environment was presented by Ng et al. (2023). The movement of vessels in a narrow channel, considering navigation rules to improve efficiency and safety through cellular automata methods, was explored by Chen L. et al. (2023). The use of deep learning for developing models for the identification and characterization of seafloor features was discussed by Lundine et al. (2023). The applied models include three convolutional neural network architectures and a generative adversarial network.

Cryptography employs cellular automata for creating secure ciphers, while in medicine (El Yacoubi & El Jai, 2002), they assist in modeling disease spread and developing effective treatment strategies. In manufacturing technologies, cellular automata are used to optimize and model processes in production, depending on the specific task, including various rules for changing cell states and conditions for their execution. The work of Ion et al. (2023) is dedicated to developing a new method for breast cancer detection in mammograms by combining CA with fuzzy logic, resulting in the creation of a so-called Fuzzy Cellular Automaton.

A detailed review of Transport Cellular Automata (TCA) models, which are effective microscopic models of traffic movement, was provided by Maerivoet & De Moor (2005). TCA models emerge from statistical mechanics and aim to reproduce macroscopic behavior with minimal description of microscopic interactions.

Research reveals the significant potential of cellular automata in modeling and understanding various physical phenomena. Combined with modern computational technologies, this allows for faster and more efficient modeling of various processes, such as metal solidification (Tang et al., 2023). Neural Cellular Automata (NCA) can learn based on diverse microstructural modeling data, including information obtained from phase fields, highlighting their flexibility and applicability in different conditions. The use of lattice gas cellular automata (LGCA) for modeling cell migration in biological environments (Hatzikirou & Deutsch, 2008), demonstrates the possibility of precise mathematical analysis and application in clinical studies. This research opens up opportunities for utilizing LGCA to model cell movement in various biological contexts, including glioma cell invasion.

Considerable attention has been devoted to the analysis of the computational complexity of cellular automata. Svyetlichnyy (2023) analyzed frontal FCAs compared to classical CAs and points out their low computational complexity and time efficiency. The use of parallel computing can significantly accelerate computations. The utilization of graphics processing units (GPUs) for accelerating CA simulations in various scientific fields has been investigated by Cagigas-Muñiz et al. (2022). Optimization techniques for memory-dependent CAs on GPU, such as stencil computational frameworks, lookup tables, and packet encoding, have been proposed, significantly improving performance compared to baseline implementations. The technique of converting irreversible functions into reversible ones to enhance energy efficiency was employed by Jaiswal et al. (2023). The authors explored the use of Quantum-dot Cellular Automata (QCA) to create and simulate regenerative full adders and full subtractors. New symmetric and planar designs for multiple inputs were proposed, surpassing existing ones in several parameters.

Advantages and disadvantages of CA

CA is a computational method used to simulate complex systems. This method has the following advantages:

- **Simplicity:** CA is based on simple rules applied to discrete cells, making them relatively easy to understand and implement.
- **Parallelism:** CA can be highly parallelized, which means they can be efficiently simulated on parallel computing architectures, leading to fast computation speeds.
- **Emergent Behavior:** CA can exhibit complex emergent behavior from simple rules. This property makes them useful for modeling and understanding systems where complexity arises from the interactions of simple components.
- **Versatility:** CA can be applied to a wide range of fields, including physics, biology, computer science, and social sciences, making them versatile tools for modeling various phenomena.
- **Self-Organization:** CA often demonstrates self-organization, where complex patterns and structures emerge spontaneously from the interactions of cells without centralized control. This property is useful for modeling self-organizing systems in nature.
- **FEM integration:** By integrating cellular automata with finite element methods, it's possible to achieve higher accuracy in simulations. One CA can be assumed as one finite element. Therefore, no additional meshing technique is required, even for complex geometry representations.

Despite the wide range of advantages, a list of disadvantages can be formulated:

- **Boundary Effects:** The behavior of CA can be sensitive to boundary conditions, leading to boundary effects that may distort simulation results, especially in finite grid environments. The solution is to use periodical boundary conditions but their proper use requires a lot of effort.

- **Rule Complexity:** While CA is based on simple rules, defining appropriate rules for a given system can be challenging, especially for systems with complex behaviors.
- **Computational Complexity:** Simulating cellular automata with large grid sizes or complex rules can be computationally intensive, requiring significant computational resources and time.
- **Interpretability:** Despite their simplicity, cellular automata models can produce complex and intricate patterns that may be difficult to interpret or analyze, especially in higher-dimensional spaces.
- **Limited Accuracy:** Cellular automata models are often simplifications of real-world systems and may not capture all relevant aspects accurately. This limitation can affect the accuracy of predictions and interpretations derived from these models.
- **Discretion:** Process discretization may not produce the desired results for modeling certain processes. Not taking into account the features of discretization can significantly increase the errors in calculations.

In summary, cellular automata offer a powerful framework for simulating complex systems and understanding emergent phenomena. However, they also have limitations related to boundary effects, computational intensity, and interpretability, which should be taken into account when applying them to real-world problems.

The computational cost of the CA method

The need to address computational expenses related to the cellular automata method is evident. Although the CA method presents flexibility and potential for modeling intricate systems, its execution can incur substantial computational costs. Creating extensive microstructures or simulating detailed mechanical characteristics of materials might require considerable computational resources and time. Furthermore, the intricacy of CA rules and the necessity for precise parameter adjustments can result in heightened computational expenditures. Hence, researchers must weigh the benefits of the CA method against its computational requirements and investigate optimization tactics to improve effectiveness.

There is no common point of view about algorithm complexity in big-O notation. In the following paragraph authors' point of view is provided.

In the case of n-dimensional grid $n \times n \times \ldots \times n$ with n_{it} as number iteration (time), the naive algorithm can be represented by the next pseudo-code:

```
for (int i = 0; i < n_it; i++)
// cycle truth $n_{it}$ iteration
  for (int j = 0; j < n; j++)
  // cycles over the grid
   for (int k = 0; k < n; k++)
     …
     for (int l = 0; l < n; l++)
     {
        check_neighbors(j, k, …,l);
        update_state(j, k, …,l);
     }
```

Such a naive implementation can be assumed as the worst case, which leads to $O(n_{it} \times n^d)$ complexity, where d is the number of dimensions.

The main idea of how to speed up the naive version of the CA algorithm is in replacement iteration over the grid by iteration over the list of cells, which can change their state. This is a very effective technique because, at the initial steps, there is a lot of free space on the field. The same situation occurs in the ending steps when only a few cells can change their state. This idea can be implemented in the next pseudo-code:

```
for (int i = 0; i < n_it; i++)
// cycle truth $n_{it}$ iteration
  {
  for (coord c : cell_list)
  // cycle over cell list
  {
     check_neighbors(c.x, c.y, …,c.z);
    update_state(c.x, c.y, …,c.z);
  }
  update_cell_list();
}
```

This approach has only two cycles with $O(n_{it} \times n)$ complexity. But the *update_cell_list*() function can be very complex because it has to predict all cells which are going to change state in the next iteration. The complexity of this function depends on solved problems, modeling techniques, etc.

Summarizing, this section can be lower in the higher estimation of CA complexity. The true performance lead between $O(n_{it} \times n)$ and $O(n_{it} \times n^d)$.

The future application of cellular automata method

The advancement of information technology and algorithms to generate statistically comparable microstructures across a range of material types, such as composite, biological, and mesoporous (aerogels) materials, is pivotal in modern materials science research focusing on material microstructure studies. The primary focus in this direction is on the development of methods for generating statistically equivalent microstructure of materials using probabilistic cellular automata.

Further development of the CA method is expected, enabling the modeling of composite, polycrystalline, and mesoporous materials in three-dimensional space. This will improve the accuracy of material modeling, and the use of parallel computing methods will accelerate the generation of large microstructures. Methods for determining the averaged characteristics of composite and mesoporous materials will continue to evolve. They will consider the anisotropy of mechanical and thermal properties of the representative volume, as well as determine its characteristics as probabilistic quantities. The use of such methods will speed up calculations compared to classical direct modeling methods.

These research directions open up new perspectives for the application of the cellular automata method in the study and analysis of material microstructures, which has significant potential to improve the quality and speed of modeling various materials.

Conclusions

Cellular automata have been explored from various perspectives, and there are still many other topics that are either under investigation or already have an extensive literature, warranting a separate review. Some of these topics, such as modeling in physics, asynchronous cellular automata, and cellular neural networks, have been highlighted in this work. Significant attention has been devoted to works in the fields of image and pattern recognition, cryptography, analysis, computer and material science. The section related to the use of CAs for material modeling encompasses the specific achievements by the authors in this domain.

Acknowledgments

This work has been supported by the Ministry of Education and Science of Ukraine in the framework of the realization of the research project “Algorithms, models, and tools of artificial intelligence for two-level modeling of complex materials behavior for dual-use technologies” (State Reg. Num. 0124U000450).

References

Aburas, M.M., Ho, Y.M., Ramli, M.F., & Ash’aari, Z.H. (2016). The simulation and prediction of spatio-temporal urban growth trends using cellular automata models: A review. *International Journal of Applied Earth Observation and Geoinformation*, *52*, 380–389. https://doi.org/10.1016/j.jag.2016.07.007.

Aladyev, V. (1974). Survey of research in the theory of homogeneous structures and their applications. *Mathematical Biosciences*, *22*, 121–154. https://doi.org/10.1016/0025-5564(74)90088-1.

Amoroso, S., & Cooper, G. (1971). Tessellation structures for reproduction of arbitrary patterns. *Journal of Computer and System Sciences*, *5*(5), 455–464. https://doi.org/10.1016/S0022-0000(71)80009-0.

Aruldoss, J.A., & Pricilla, J.A. (2014). Mathematical model of cellular automata and fractals in cancer growth. *International Journal of Computing Algorithm*, *3*(3), 243–245.

Aso, H., & Honda, N. (1985). Dynamical characteristics of linear cellular automata. *Journal of Computer and System Sciences*, *30*(3), 291–317. https://doi.org/10.1016/0022-0000(85)90048-0.

Banks, E.R. (1970). Universality in cellular automata. In *11th Annual Symposium on Switching and Automata Theory (swat 1970)*. IEEE. 194–215. https://doi.org/10.1109/SWAT.1970.27.

Baran, K., Sitko, M., & Madej, L. (2024). Analysis of the influence of a pseudo-random number generator type on the kinetics of the cellular automata recrystallization model. *Journal of Computer Science*, *75*, 102193. https://doi.org/10.1016/j.jocs.2023.102193.

Bardell, P.H. (1990). Analysis of cellular automata used as pseudorandom pattern generators. In *Proceedings. International Test Conference*. IEEE. 762–768. https://doi.org/10.1109/TEST.1990.114093.

Bhardwaj, R., & Bhagat, D. (2018). Two level encryption of grey scale image through 2D cellular automata. *Procedia Computer Science*, *125*, 855–861. https://doi.org/10.1016/j.procs.2017.12.109.

Bouchkaren, S., & Lazaar, S. (2014). A fast cryptosystem using reversible cellular automata. *International Journal of Advanced Computer Science and Applications*, *5*(5), 207–210. https://doi.org/10.14569/IJACSA.2014.050531.

Brady, M.L., Raghavan, R., & Slawny, J. (1989). Probabilistic cellular automata in pattern recognition. In *International 1989 Joint Conference on Neural Networks*, *1*. IEEE, 177–182. https://doi.org/10.1109/IJCNN.1989.118577.

Braga, G., Cattaneo, G., Flocchini, P., & Vogliotti, C.Q. (1995). Pattern growth in elementary cellular automata. *Theoretical Computer Science*, *145*(1–2), 1–26. https://doi.org/10.1016/0304-3975(94)00155-C.

Bruno, M. (1994). Inherent generation of fractals by cellular automata. *Complex Systems*, 8(5), 347–366.

Burks, W. (1970). *Essays on Cellular Automata.* University of Illinois Press.

Cagigas-Muñiz, D., Diaz-del-Rio, F., Sevillano-Ramos, J.L., & Guisado-Lizar, J.-L. (2022). Efficient simulation execution of cellular automata on GPU. *Simulation Modelling Practice and Theory*, *118*, 102519. https://doi.org/10.1016/j.simpat.2022.102519.

Chang, J.H., Ibarra, O.H., & Vergis, A. (1986). On the power of one-way communication. *27th Annual Symposium on Foundations of Computer Science (sfcs 1986)*. IEEE. 455–464. https://doi.org/10.1109/SFCS.1986.37.

Chaudhuri, P.P., Chowdhury, D.R., Nandi, S., & Chattopadhyay, S. (1997). *Additive Cellular Automata: Theory and Applications* (vol. 1). IEEE Computer Society Press.

Chen, L., Qi, J., & Shi, J. (2023). A cellular automata ship traffic flow model considering navigation rules in narrowing channel. *Alexandria Engineering Journal*, *69*, 715–726. https://doi.org/10.1016/j.aej.2023.01.062.

Chen, M.-S., Yuan, W.-Q., Lin, Y.C., Li, H.-B., & Zou, Z.-H. (2017). Modeling and simulation of dynamic recrystallization behavior for 42CrMo steel by an extended cellular automaton method. *Vacuum*, *146*, 142–151. https://doi.org/10.1016/j.vacuum.2017.09.041.

Codd, E.F. (1968). *Cellular Automata.* Academic Press.

Connelly, R. (1986). Winning Ways for Your Mathematical Plays, Volumes I & II By Elwyn R. Berlekamp, John H. Conway. *The American Mathematical Monthly*, *93*(5), 411–414. https://doi.org/10.1080/00029890.1986.11971843.

Conway, J.H. (1976). *On Numbers and Games.* Academic Press.

Cui, J., Li, B., Liu, Z., Xiong, Y., Qi, F., Zhao, Z., & Zhu, S. (2023). Numerical investigation of grain structure under the rotating arc based on cellular automata-finite element method during vacuum arc remelting process. *Metallurgical and Materials Transactions B*, *54*, 661–672. https://doi.org/10.1007/s11663-022-02716-x.

Culik II, K., & Dube, S. (1991). An efficient solution of the firing mob problem. *Theoretical Computer Science*, *91*(1), 57–69. https://doi.org/10.1016/0304-3975(91)90267-6.

Culik II, K., & Yu, S. (1988). Undecidability of CA classification schemes. *Complex Systems*, *2*(2), 177–190.

Culik II, K., Hurd, L.P., & Yu, S. (1990). Computation theoretic aspects of cellular automata. *Physica D: Nonlinear Phenomena*, *45*(1–3), 357–378. https://doi.org/10.1016/0167-2789(90)90194-T.

D'amico, M., Manzini, G., & Margara, L. (2003). On computing the entropy of cellular automata. *Theoretical Computer Science*, *290*(3), 1629–1646. https://doi.org/10.1016/S0304-3975(02)00071-3.

Das, S., Mukherjee, S., Naskar, N., & Sikdar, B.K. (2009). Modeling single length cycle nonlinear cellular automata for pattern recognition. In *2009 World Congress on Nature & Biologically Inspired Computing (NaBIC)* (pp. 198–203). https://doi.org/10.1109/NABIC.2009.5393685.

Deng, X., Chen, H., Xu, Q., Feng, F., Chen, X., Lv, X., Lin, X., & Fu, T. (2022). Topology optimization design of three-dimensional multi-material and multi-body structure based on irregular cellular hybrid cellular automata method. *Scientific Reports*, *12*(1), 5602. https://doi.org/10.1038/S41598-022-09249-Y.

Dennunzio, A., Formenti, E., & Margara, L. (2024). An efficient algorithm deciding chaos for linear cellular automata over (Z\mZ)n with applications to data encryption. *Information Sciences*, *657*, 119942. https://doi.org/10.1016/j.ins.2023.119942.

Ding, R., & Guo, Z.X. (2002). Microstructural modelling of dynamic recrystallisation using an extended cellular automaton approach. *Computational Materials Science*, *23*(1–4), 209–218. https://doi.org/10.1016/s0927-0256(01)00211-7.

Dioșan, L., Andreica, A., & Enescu, A. (2017). The use of simple cellular automata in image processing. *Studia Universitatis Babeș-Bolyai Informatica*, *62*(1), 5–14. https://doi.org/10.24193/subbi.2017.1.01.

Dobravec, T., Mavrič, B., & Šarler, B. (2017). A cellular automaton – finite volume method for the simulation of dendritic and eutectic growth in binary alloys using an adaptive mesh refinement. *Journal of Computational Physics*, *349*, 351–375. https://doi.org/10.1016/j.jcp.2017.08.011.

Dyer, Ch.R. (1980). One-way bounded cellular automata. *Information and Control*, *44*(3), 261–281. https://doi.org/10.1016/S0019-9958(80)90164-3.

Efstathiou, G., Gkantonas, S., Giusti, A., Mastorakos, E., Foale, C.M., & Foale, R.R. (2023). Simulation of the December 2021 Marshall fire with a hybrid stochastic Lagrangian-cellular automata model. *Fire Safety Journal*, *138*, 103795. https://doi.org/10.1016/j.firesaf.2023.103795.

El Yacoubi, S., & El Jai, A. (2002). Cellular automata modelling and spreadability. *Mathematical and Computer Modelling*, *36*(9–10), 1059–1074. https://doi.org/10.1016/S0895-7177(02)00259-5.

Finelli, M., Manzini, G., & Margara, L. (1998). Lyapunov exponents versus expansivity and sensitivity in cellular automata. *Journal of Complexity*, *14*(2), 210–233. https://doi.org/10.1006/jcom.1998.0474.

Fúster-Sabater, A., & Caballero-Gil, P. (2009). Synthesis of cryptographic interleaved sequences by means of linear cellular automata. *Applied Mathematics Letters*, *22*(10), 1518–1524. https://doi.org/10.1016/j.aml.2009.03.018.

Gács, P. (1986). Reliable computation with cellular automata. *Journal of Computer and System Sciences*, *32*(1), 15–78. https://doi.org/10.1016/0022-0000(86)90002-4.

Garzon, M. (1995). *Models of Massive Parallelism: Analysis of Cellular Automata and Neural Networks.* Springer Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-77905-3.

Ghosh, P., Mukhopadhyay, A., Chanda, A., Mondal, P., Akhand, A., Mukherjee, S., Nayak, S.K., Ghosh, S., Mitra, D., Ghosh, T., & Hazra, S. (2017). Application of cellular automata and Markov-chain model in geospatial environmental modeling – A review. *Remote Sensing Applications: Society and Environment*, *5*, 64–77. https://doi.org/10.1016/j.rsase.2017.01.005.

Goetz, R.L., & Seetharaman, V. (1998). Modeling dynamic recrystallization using cellular automata. *Scripta Materialia*, *38*(3), 405–413. https://doi.org/10.1016/s1359-6462(97)00500-9.

Green, F. (1987). NP-complete problems in cellular automata. *Complex Systems*, *1*(3), 453–474.

Guan, P. (1987). Cellular automaton public-key cryptosystems. *Complex Systems*, *1*(1), 51–57.

Gutowitz, H. (1993). Cryptography with dynamical systems. In N. Boccara, E. Goles, S. Martinez, P. Picco (Eds.), *Cellular Automata and Cooperative Phenomena* (pp. 237–274). Springer Dordrecht. https://doi.org/10.1007/978-94-011-1691-6_21.

Gütschow, J., Nesme, V., & Werner, R.F. (2010). The fractal structure of cellular automata on abelian groups. In *Automata 2010 – 16th Intl. Workshop on CA and DCS. Discrete Mathematics & Theoretical Computer Science* (pp. 51–70). https://doi.org/10.46298/dmtcs.2759.

Hallberg, H., Wallin, M., & Ristinmaa, M. (2010). Simulation of discontinuous dynamic recrystallization in pure Cu using a probabilistic cellular automaton. *Computational Materials Science*, *49*(1), 25–34. https://doi.org/10.1016/j.commatsci.2010.04.012.

Harao, M., & Noguchi, S. (1978). On some dynamical properties of finite cellular automata. *IEEE Transactions on Computers*, *C-27*(1), 42–52. https://doi.org/10.1109/TC.1978.1674951.

Hatzikirou, H., & Deutsch, A. (2008). Cellular automata as microscopic models of cell migration in heterogeneous environments. *Current Topics in Developmental Biology*, *81*, 401–434. https://doi.org/10.1016/S0070-2153(07)81014-3.

Hesselbarth, H.W., & Göbel, I.R. (1991). Simulation of recrystallization by cellular automata. *Acta Metallurgica et Materialia*, *39*(9), 2135–2143. https://doi.org/10.1016/0956-7151(91)90183-2.

Holland, J.H. (1976). Studies of the spontaneous emergence of self-replicating systems using cellular automata and formal grammers. In A. Lindenmayer, G. Rozenberg (Eds.), *Automata, Languages, Development* (pp. 385–404). North-Holland Publishing.

Hurd, L.P., Kari, J., & Culik, K. (1992). The topological entropy of cellular automata is uncomputable. *Ergodic Theory and Dynamical Systems*, *12*(2), 255–265. https://doi.org/10.1017/S0143385700006738.

Ibarra, O.H., & Kim, S.M. (1984). Characterizations and computational complexity of systolic trellis automata. *Theoretical Computer Science*, *29*(1–2), 123–153. https://doi.org/10.1016/0304-3975(84)90015-X.

Ion, I.-A., Moroz-Dubenco, C., & Andreica, A. (2023). Breast cancer images segmentation using fuzzy cellular automaton. *Procedia Computer Science*, *225*, 999–1008. https://doi.org/10.1016/j.procs.2023.10.087.

Isa, M.A.M., Ahmad, M.M., Sani, N.F.M., Hashim, H., & Mahmod, R. (2014). Cryptographic key exchange protocol with message authentication codes (MAC) using finite state machine. *Procedia Computer Science*, *42*, 263–270. https://doi.org/10.1016/j.procs.2014.11.061.

Itô, M., Ôsato, N., & Nasu, M. (1983). Linear cellular automata over Z_m. *Journal of Computer and System Sciences*, *27*(1), 125–140. https://doi.org/10.1016/0022-0000(83)90033-8.

Jaiswal, R.K., Singh, D., Sharma, V.M., & Vishwakarma, D. (2023). Performance analysis of full adder and full subtractor using quantum-dot cellular automata. *e-Prime – Advances in Electrical Engineering, Electronics and Energy*, *6*, 100279. https://doi.org/10.1016/j.prime.2023.100279.

Jeon, J.-Ch. (2010). Non-linear and non-group cellular automata chaining technique for cryptographic applications. *Mathematical and Computer Modelling*, *51*(7–8), 995–999. https://doi.org/10.1016/j.mcm.2009.08.033.

Jump, J.R., & Kirtane, J.S. (1974). On the interconnection structure of cellular networks. *Information and Control*, *24*(1), 74–91. https://doi.org/10.1016/S0019-9958(74)80024-0.

Kari, L., Rozenberg, G., & Salomaa, A. (1997). L systems. In Rozenberg, G., Salomaa, A. (Eds.), *Handbook of Formal Languages* (Vol. 1: *Word, Language, Grammar*, pp. 253–328). Springer Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-59136-5_5.

Khedmati, Y., Parvaz, R., & Behroo, Y. (2020). 2D Hybrid chaos map for image security transform based on framelet and cellular automata. *Information Sciences*, *512*, 855–879. https://doi.org/10.1016/j.ins.2019.10.028.

Kosaraju, S.R. (1975). Speed of recognition of context-free languages by array automata. *SIAM Journal on Computing*, *4*(3), 331–340. https://doi.org/10.1137/0204028.

Krishna, V.M., Mishra, S., & Gurrala, J. (2022). Cellular automata and their applications: a review. *International Journal of Computer Applications*, *184*(5), 13–23. https://doi.org/10.5120/ijca2022922010.

Kurka, P. (1997). Languages, equicontinuity and attractors in cellular automata. *Ergodic Theory and Dynamical Systems*, *17*(2), 417–433. https://doi.org/10.1017/S014338579706985X.

Langton, Ch.G. (1984). Self-reproduction in cellular automata. *Physica D: Nonlinear Phenomena*, *10*(1–2), 134–144. https://doi.org/10.1016/0167-2789(84)90256-2.

Lee, W., Bae, J., Lee, H., Kang, S., Kang, S.-H., & Yoon, J. (2022). Numerical simulation of dendritic growth and porosity evolution in solidification of Al-Cu alloy with lattice Boltzmann – cellular automata method. *Journal of Alloys and Compounds*, *929*, 167233. https://doi.org/10.1016/j.jallcom.2022.167233.

Lewis, A.E., Seckler, M.M., Kramer, H., & van Rosmalen, G. (2015). *Industrial Crystallization: Fundamentals and Applications*. Cambridge University Press. https://doi.org/10.1017/CBO9781107280427.

Li, H., Sun, X., & Yang, H. (2016). A three-dimensional cellular automata-crystal plasticity finite element model for predicting the multiscale interaction among heterogeneous deformation, DRX microstructural evolution and mechanical responses in titanium alloys. *International Journal of Plasticity*, *87*, 154–180. https://doi.org/10.1016/j.ijplas.2016.09.008.

Lindenmayer, A. (1968). Mathematical models for cellular interactions in development. Parts I and II. *Journal of Theoretical Biology*, *18*(3), 280–315. https://doi.org/10.1016/0022-5193(68)90079-9.

Litow, B., & Dumas, Ph. (1993). Additive cellular automata and algebraic series. *Theoretical Computer Science*, *119*(2), 345–354. https://doi.org/10.1016/0304-3975(93)90165-P.

Liu, S., Hong, K.-m., & Shin, Y.C. (2021). A novel 3D cellular automata-phase field model for computationally efficient dendrite evolution during bulk solidification. *Computational Materials Science*, *192*, 110405. https://doi.org/10.1016/j.commatsci.2021.110405.

Lundine, M.A., Brothers, L.L., & Trembanis, A.C. (2023). Deep learning for pockmark detection: Implications for quantitative seafloor characterization. *Geomorphology*, *421*, 108524. https://doi.org/10.1016/j.geomorph.2022.108524.

Madej, L., & Sitko, M. (2022). Computationally efficient cellular automata-based full-field models of static recrystallization: A perspective review. *Steel Research International*, *94*(3), 2200657. https://doi.org/10.1002/srin.202200657.

Maerivoet, S., & De Moor, B. (2005). Cellular automata models of road traffic. *Physics Reports*, *419*(1), 1–64. https://doi.org/10.1016/j.physrep.2005.08.005.

Mahajan, M. (1992). *Studies in language classes defined by different types of time-varying cellular automata*. Ph.D. Thesis, Indian Institute of Technology, Madras, India.

Maiti, S., Sengupta, M., & Chowdhury, D.R. (2021). Generating nonlinear codes for multi-bit symbol error correction using cellular automata. *Physica D: Nonlinear Phenomena*, *415*, 132758. https://doi.org/10.1016/j.physd.2020.132758.

Maji, P., & Chaudhuri, P.P. (2005). Fuzzy cellular automata for modeling pattern classifier. *IEICE Transactions on Information and Systems*, *E88-D*(4), 691–702. https://doi.org/10.1093/ietisy/e88-d.4.691.

Maji, P., Shaw, C., Ganguly, N., Sikdar, B. K., & Chaudhuri, P.P. (2003). Theory and application of cellular automata for pattern classification. *Fundamenta Informaticae*, *58*(3–4), 321–354.

Manzini, G., & Margara, L. (1999). Attractors of linear cellular automata. *Journal of Computer and System Sciences*, *58*(3), 597–610. https://doi.org/10.1006/jcss.1998.1609.

Martin, O., Odlyzko, A.M., & Wolfram, S. (1984). Algebraic properties of cellular automata. *Communications in Mathematical Physics*, *93*, 219–258. https://doi.org/10.1007/BF01223745.

Mastorakos, E., Gkantonas, S., Efstathiou, G., & Giusti, A. (2023). A hybrid stochastic Lagrangian – cellular automata framework for modelling fire propagation in inhomogeneous terrains. *Proceedings of the Combustion Institute*, *39*(3), 3853–3862. https://doi.org/10.1016/j.proci.2022.07.240.

Mi, J., Hou, H., Zhang, S., Hua, Y., Yang, Y., Zhu, Y., & Ding, Z. (2023). Detecting long-term effects of mining-induced ground deformation on plant succession in semi-arid areas using a cellular automata model. *Ecological Indicators*, *151*, 110290. https://doi.org/10.1016/j.ecolind.2023.110290.

Mohamed, F.K. (2014). A parallel block-based encryption schema for digital images using reversible cellular automata. *Engineering Science and Technology, an International Journal*, *17*(2), 85–94. https://doi.org/10.1016/j.jestch.2014.04.001.

Morita, K. (1992). Computation-universality of one-dimensional one-way reversible cellular automata. *Information Processing Letters*, *42*(6), 325–329. https://doi.org/10.1016/0020-0190(92)90231-J.

Myhill, J. (1963). The converse of Moore's Garden-of-Eden theorem. *Proceedings of the American Mathematical Society*, *14*, 685–686. https://doi.org/10.1090/S0002-9939-1963-0155764-9.

Neumann, J., von (1951). The general and logical theory of automata. In L.A. Jeffress (Ed.), *Cerebral Mechanisms in Behavior. The Hixon Symposium* (pp. 1–41). John Wiley & Sons, Chapman & Hall.

Neumann, J., von (1966). *Theory of Self Reproducing Automata*. University of Illinois Press.

Ni, T. (2003). Some properties of fractals generated by linear cellular automata. *Tsinghua Science and Technology*, *8*(5), 557–563.

Niemi, V. (1997). Cryptology: language-theoretic aspects. In Rozenberg, G., Salomaa, A. (Eds.), *Handbook of Formal Languages* (Vol. 2: *Linear Modeling: Background and Application*, pp. 507–524). Springer Berlin, Heidelberg. https://doi.org/10.1007/978-3-662-07675-0_11.

Nishio, H., & Kobuchi, Y. (1975). Fault tolerant cellular spaces. *Journal of Computer and System Sciences*, *11*(2), 150–170. https://doi.org/10.1016/S0022-0000(75)80065-1.

Packard, N.H., & Wolfram, S. (1985). Two-dimensional cellular automata. *Journal of Statistical Physics*, *38*, 901–946. https://doi.org/10.1007/BF01010423.

Pesavento, U. (1995). An implementation of von Neumann's self-reproducing machine. *Artificial Life*, *2*(4), 337–354. https://doi.org/10.1162/artl.1995.2.4.337.

Popova, E., Staraselski, Y., Brahme, A., Mishra, R.K., & Inal, K. (2015). Coupled crystal plasticity – Probabilistic cellular automata approach to model dynamic recrystallization in magnesium alloys. *International Journal of Plasticity*, *66*, 85–102. https://doi.org/10.1016/J.IJPLAS.2014.04.008.

Popovici, A., & Popovici, D. (2002). Cellular automata in image processing. *International Symposium on the Mathematical Theory of Networks and Systems*.

Prusinkiewicz, P., & Lindenmayer, A. (1990). *The algorithmic beauty of plants*. Springer New York, NY. https://doi.org/10.1007/978-1-4613-8476-2.

Qian, M., & Guo, Z.X. (2004). Cellular automata simulation of microstructural evolution during dynamic recrystallization of an HY-100 steel. *Materials Science and Engineering: A*, *365*(1–2), 180–185. https://doi.org/10.1016/J.MSEA.2003.09.025.

Raabe, D., & Becker, R. (2000). Coupling of a crystal plasticity finite-element model with a probabilistic cellular automaton for simulating primary static recrystallization in aluminium. *Modelling and Simulation in Materials Science and Engineering*, *8*(4), 445. https://doi.org/10.1088/0965-0393/8/4/304.

Reuther, K., & Rettenmayr, M. (2014). Perspectives for cellular automata for the simulation of dendritic solidification – A review. *Computational Materials Science*, *95*, 213–220. https://doi.org/10.1016/j.commatsci.2014.07.037.

Richardson, D. (1972). Tessellations with local transformations. *Journal of Computer and System Sciences*, *6*(5), 373–388. https://doi.org/10.1016/S0022-0000(72)80009-6.

Róka, Z. (1994). One-way cellular automata on Cayley graphs. *Theoretical Computer Science*, *132*(1–2), 259–290. https://doi.org/10.1016/0304-3975(94)90236-4.

Rosin, P.L. (2005). Training cellular automata for image processing. In H. Kalviainen, J. Parkkinen, A. Kaarna (Eds.), *Image Analysis 14th Scandinavian Conference, SCIA 2005, Joensuu, Finland, June 19–22, 2005, Proceedings* (pp. 195–204). Springer Berlin, Heidelberg. https://doi.org/10.1007/11499145_22.

Roy, S., Karjee, J., Rawat, U.S., Pratik, D.N., & Dey, N. (2016). Symmetric key encryption technique: A cellular automata based approach in wireless sensor networks. *Procedia Computer Science*, *78*, 408–414. https://doi.org/10.1016/j.procs.2016.02.082.

Sabau, A.S., Yuan, L., Fattebert, J.-L., & Turner, J.A. (2023). An OpenMP GPU-offload implementation of a non-equilibrium solidification cellular automata model for additive manufacturing. *Computer Physics Communications*, *284*, 108605. https://doi.org/10.1016/j.cpc.2022.108605.

Sadiq, M.D., & Kumar, B.H. (2015). Efficient cryptography using cellular automata rules. *International Journal of Emerging Engineering Research and Technology*, *3*(12), 18–25.

Salo, V., Theyssier, G., & Törmä, I. (2022). Cellular automata and bootstrap percolation. *Theoretical Computer Science*, *924*, 34–45. https://doi.org/10.1016/j.tcs.2022.04.015.

Sapino, F., Haer, T., Saiz-Santiago, P., & Pérez-Blanco C.D. (2023). A multi-agent cellular automata model to explore water trading potential under information transaction costs. *Journal of Hydrology*, *618*, 129195. https://doi.org/10.1016/j.jhydrol.2023.129195.

Sarkar, P. (2000). A brief history of cellular automata. *ACM Computing Surveys*, *32*(1), 80–107. https://doi.org/10.1145/349194.349202.

Sarkar, P., & Barua, R. (1998). Multidimensional σ-automata, π-polynomials and generalised S-matrices. *Theoretical Computer Science*, *197*(1–2), 111–138. https://doi.org/10.1016/S0304-3975(97)00160-6.

Seiferas, J.I. (1974). Observations on nondeterministic multidimensional iterative arrays. In *Proceedings of the ACM Symposium on the Theory of Computing* (pp. 276–289). ACM Press. https://doi.org/10.1145/800119.803905.

Sethi, B., & Das, S. (2016). On the use of asynchronous cellular automata in symmetric−key cryptography. *Computing and Communications*, 30–41.

Smith III, A.R. (1972). Real-time language recognition by one-dimensional cellular automata. *Journal of Computer and System Sciences*, *6*(3), 233–253.

Smith III, A.R. (1976). *Introduction to and Survey of Polyautomata Theory*. In A. Lindenmayer, G. Rozenberg (Eds.), *Automata, Languages, Development* (pp. 405–422). North-Holland Publishing.

Sree, P.K., & Babu, I.R. (2008). Identification of protein coding regions in genomic DNA using unsupervised FMACA based pattern classifier. *IJCSNS International Journal of Computer Science and Network Security*, *8*(1), 305–309. https://doi.org/10.48550/arXiv.1401.6484.

Sree, P.K., & Babu, I.R. (2010). Identification of promoter region in genomic DNA using cellular automata based text clustering. *The International Arab Journal of Information Technology*, *7*(1), 75–78.

Sree, P.K., Babu, I.R., & Devi, N.U. (2013). PSMACA: An automated protein structure prediction using MACA (Multiple Attractor Cellular Automata). *Journal of Bioinformatics and Intelligent Control*, *2*(3), 211–215. https://doi.org/10.1166/jbic.2013.1052.

Sree, P.K., Babu, I.R., & Devi, N.U (2014). Cellular automata and its applications in bioinformatics: A review. *Global Perspectives on Artificial Intelligence*, *2*(2), 16–22. https://doi.org/10.48550/arXiv.1404.0453.

Sutner, K. (1989). A note on the Culik-Yu classes. *Complex Systems*, *3*(1), 107–115.

Sutner, K. (1990). Classifying circular cellular automata. *Physica D: Nonlinear Phenomena*, *45*(1–3), 386–395. https://doi.org/10.1016/0167-2789(90)90196-V.

Svyetlichnyy, D. (2023). Frontal cellular automata for modelling microstructure evolution: Computational complexity analysis. *Computational Materials Science*, *230*, 112478. https://doi.org/10.1016/j.commatsci.2023.112478.

Tang, J., Kumar, S., De Lorenzis, L., & Hosseini, E. (2023). Neural cellular automata for solidification microstructure modelling. *Computer Methods in Applied Mechanics and Engineering*, *414*, 116197. https://doi.org/10.1016/j.cma.2023.116197.

Terry-Jack, M., & O'Keefe, S. (2023). Classifying 1D elementary cellular automata with the 0–1 test for chaos. *Physica D: Nonlinear Phenomena*, *453*, 133786. https://doi.org/10.1016/j.physd.2023.133786.

Tezuka, S., & Fushimi, M. (1994). A method of designing cellular automata as pseudorandom number generators for built-in self-test for VLSI. *Contemporary Mathematics*, *168*, 363–367. https://doi.org/10.1090/conm/168/01713.

Tomassini, M., & Perrenoud, M. (2001). Cryptography with cellular automata. *Applied Soft Computing*, *1*(2), 151–160. https://doi.org/10.1016/S1568-4946(01)00015-1.

Ushmaev, D., Liao, Z., Notron, A., & Axinte, D. (2022). On the importance of interface stability in cellular automata models: planar and dendritic solidification in laser melted YSZ. *Materials & Design*, *219*, 110823. https://doi.org/10.1016/j.matdes.2022.110823.

Vispoel, M., Daly, A.J., & Baetens, J.M. (2022). Progress, gaps and obstacles in the classification of cellular automata. *Physica D: Nonlinear Phenomena*, *432*, 133074. https://doi.org/10.1016/j.physd.2021.133074.

Vitányi, P. (1973). Sexually reproducing cellular automata. *Mathematical Biosciences*, *18*(1–2), 23–54. https://doi.org/10.1016/0025-5564(73)90018-7.

Vodka, O. (2019). Analysis of quantitative characteristics of microstructures that are generated by the probabilistic cellular automata method. In *2019 IEEE 2nd Ukraine Conference on Electrical and Computer Engineering, UKRCON 2019 – Proceedings* (pp. 990–994). https://doi.org/10.1109/UKRCON.2019.8879959.

Vodka, O. (2020). Computer modeling of microstructures with probabilistic cellular automata method using different nucleation rate functions. In S. Subbotin (Ed.), *International Workshop on Computer Modeling and Intelligent Systems (CMIS-2020). Zaporizhzhia, Ukraine, April 27 – May 1, 2020* (pp. 450–461). https://doi.org/10.32782/cmis/2608-34.

Vollmar, T. (1977). Cellular spaces and parallel algorithms, an introductory survey. In M. Feilmeier (Ed.), *Parallel Computation-Parallel Mathematics. Proceedings of the IMACS (AICA)-GI Symposium, March 14–16, 1977, Technical University of Munich* (pp. 49–58). North-Holland Publishing.

Wang, L., Fang, G., & Qian, L. (2018). Modeling of dynamic recrystallization of magnesium alloy using cellular automata considering initial topology of grains. *Materials Science and Engineering: A*, *711*, 268–283. https://doi.org/10.1016/j.msea.2017.11.024.

Wang, W. (2003). *A Mathematical Model of Dendritic Microstructures in Nickel-Based Superalloys* [Doctoral thesis, Imperial College London]. Search Spiral. Imperial College London's repository. http://hdl.handle.net/10044/1/11524.

Wikipedia (n.d.). *Dendrite (metal)*, https://en.wikipedia.org/wiki/Dendrite_%28metal%29.

Willson, S.J. (1984). Cellular automata can generate fractals. *Discrete Applied Mathematics*, *8*(1), 91–99. https://doi.org/10.1016/0166-218X(84)90082-9.

Wolfram, S. (1983). Statistical mechanics of cellular automata. *Reviews of Modern Physics*, *55*(3), 601–644. https://doi.org/10.1103/RevModPhys.55.601.

Wolfram, S. (1984). Computation theory of cellular automata. *Communications in Mathematical Physics*, *96*, 15–57. https://doi.org/10.1007/BF01217347.

Wolfram, S. (1986). *Theory and Applications of Cellular Automata: Including Selected Papers 1983–1986*. World Scientific.

Wolnik, B., Dziemiańczuk, M., & De Baets, B. (2023). Non-uniform number-conserving elementary cellular automata on the infinite grid: A tale of the unexpected. *Information Sciences*, *649*, 119680. https://doi.org/10.1016/j.ins.2023.119680.

Wongthanavasu, S., & Ponkaew, J. (2013). Cellular Automata for Pattern Recognition. In A. Salcido (Ed.), *Emerging Applications of Cellular Automata* (pp. 53–67). IntechOpen. https://doi.org/10.5772/52364.

Yamada, H., & Amoroso, S. (1969). Tessellation Automata. *Information and Control*, *14*(3), 299–317. https://doi.org/10.1016/S0019-9958(69)90090-4.

Zhang, Ch., Zhang, L., Xu, Q., Xia, Y., & Shen, W. (2016). The kinetics and cellular automaton modeling of dynamic recrystallization behavior of a medium carbon Cr-Ni-Mo alloyed steel in hot working process. *Materials Science and Engineering: A*, *678*, 33–43. https://doi.org/10.1016/j.msea.2016.09.056.

Zhao, Y., Qin, R.S., & Chen, D.F. (2013). A three-dimensional cellular automata model coupled with finite element method and thermodynamic database for alloy solidification. *Journal of Crystal Growth*, *377*, 72–77. https://doi.org/10.1016/j.jcrysgro.2013.05.006.

Zhao, Y., Chen, D., Long, M., Arif, T.T., & Qin, R. (2014). A three-dimensional cellular automata model for dendrite growth with various crystallographic orientations during solidification. *Metallurgical and Materials Transactions: B*, *45*, 719–725. https://doi.org/10.1007/s11663-013-9960-3.

Zhao, Y., Qin, R., Chen, D., Wan, X., Li, Y., & Ma, M. (2015). A three dimensional cellular automata model for dendrite growth in non-equilibrium solidification of binary alloy. *Steel Research International*, *86*(12), 1490–1497. https://doi.org/10.1002/srin.201400318.