

Article

Not peer-reviewed version

On the Theory of Partial Difference Equations: From Numerical Methods to Language of Complexity

[Bik Kuang Min](#)*

Posted Date: 13 August 2025

doi: 10.20944/preprints2025071221.v2

Keywords: partial difference equations; complex systems; cellular automata; coupled map lattice; discrete dynamical systems



Preprints.org is a free multidisciplinary platform providing preprint service that is dedicated to making early versions of research outputs permanently available and citable. Preprints posted at Preprints.org appear in Web of Science, Crossref, Google Scholar, Scilit, Europe PMC.

Copyright: This open access article is published under a Creative Commons CC BY 4.0 license, which permit the free download, distribution, and reuse, provided that the author and preprint are cited in any reuse.

Disclaimer/Publisher's Note: The statements, opinions, and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions, or products referred to in the content.

Article

On the Theory of Partial Difference Equations: From Numerical Methods to Language of Complexity

Bik Kuang Min

National University of Malaysia; a209555@siswa.ukm.edu.my

Abstract

This work develops a theoretical framework for *Partial Difference Equations* (PΔE) as a natural mathematical language for modeling discrete-time, discrete-space systems. Motivated by the limitations of continuous partial differential equations (PDE) in representing inherently discrete phenomena, we begin by defining PΔE in terms of discrete function spaces and shift operators, contrasting them with ordinary difference equations (OΔE) and PDE, and clarifying the scope of our study. We then examine linear PΔE, outlining their main types, providing formal definitions, and presenting selected analytic solutions in simple cases. Building on this, we introduce the discrete functional analytic setting: discrete function spaces, Hilbert space structure, and discrete operators, including difference and shift operators, and study their algebraic and adjoint properties. The discrete Green’s function is also defined within this framework. As a demonstration of the framework’s unifying power, we reformulate a wide range of well-known discrete models, including elementary cellular automata, coupled map lattices, Conway’s Game of Life, the Abelian sandpile model, the Olami–Feder–Christensen earthquake model, forest fire models, the Ising model, the Kuramoto Firefly Model, the Greenberg–Hastings Model, and the Langton’s ant as explicit PΔE. For each case, we focus on obtaining a concise and mathematically elegant formulation rather than detailed dynamical analysis. Finally, we compare the “discrete universe” of PΔE with the continuous universe of PDE, highlighting their structural parallels and their respective connections to discrete mathematics and continuous analysis. This reveals PΔE and PDE as mathematical “twins”, analogous in form yet rooted in fundamentally different underlying mathematics. The generality of the PΔE formalism suggests broad applicability, from modeling biological and ecological processes to analyzing complex networks, emergent computation, and other spatiotemporally extended systems.

Keywords: partial difference equations; complex systems; cellular automata; coupled map lattice; discrete dynamical systems

Contents

Preface	4
1. Introduction to Partial Difference Equations	5
1.1. Motivation and Background	5
1.2. Discrete Function	5
1.3. Shift Operator	5
1.4. Partial Shift Operator	6
1.5. Ordinary Difference Equations	6
1.6. Partial Difference Equation	6
1.7. The Order of a Partial Difference Equation	7
1.8. Comparison with Ordinary Difference Equations	7
1.9. Comparison with Partial Differential Equations	8

1.10. Advantages of Rewriting Complex Systems as Partial Difference Equations	8
1.11. Scope and Limitations	8
1.12. Classification and Examples	9
1.13. Notations	9
2. Linear Partial Difference Equations	11
2.1. Motivation	11
2.2. Definition	11
2.3. Discrete 1D Transport Equation	11
2.4. Discrete 1D Diffusion Equation	13
2.5. Discrete 1D Wave Equation	14
3. Discrete Function Spaces and Operators	15
3.1. Motivation	15
3.2. Discrete Function Spaces	16
3.3. Operators	17
3.4. The Kronecker Delta Function	22
3.5. The Discrete Green’s Function	23
3.6. Boundary Value Problems	25
4. Elementary Cellular Automata	26
4.1. Introduction	26
4.2. Notation Evolution	26
4.3. The Rule 90	27
4.4. Rule 30	30
4.5. The Rule 153	31
4.6. Rule 45	32
4.7. Rule 105	33
4.8. Rule 169	34
4.9. Rule 210	35
4.10. Rule 137	36
4.11. Rule 110	37
5. Other 1D Nonlinear Partial Difference Equations	38
5.1. Introduction	38
5.2. Sierpinski Fan Equation	38
5.3. Mod 5 Sierpiński Equation	39
5.4. Mod 4 Sierpinski Fractal Equation	39
6. Coupled Map Lattice	41
6.1. 1D Coupled Map Lattice	41
6.2. 2D Coupled Map Lattice	42
6.3. Explanation	42
7. The Conway’s Game of Life	42
7.1. Conway’s Equation	42
7.2. Interpretation	43
8. Abelian Sandpile Model	43
8.1. Abelian Sandpile Equation	43
8.2. Remarks	43
8.3. The Sandpile Fractal	44

8.4. Extended Sandpile Model	44
8.5. Explanation for the Self-Organized Criticality	45
8.6. Explanation for the Power Law Distribution	46
9. OFC Earthquake Model	46
9.1. OFC Earthquake Equation	46
9.2. Observations and Phenomena	47
10. Forest Fire Model	47
10.1. Forest Fire Equation	47
10.2. Definitions	47
10.3. Transition Logic (via mod 3)	48
10.4. Interpretation	48
10.5. Observations and Phenomena	48
11. Kuramoto Firefly Model	48
11.1. Kuramoto Equation	48
11.2. Principle of Locality	49
11.3. Observations and Phenomena	49
12. Ising Model	50
12.1. Governing Equation	50
12.2. Physical Interpretation	51
13. Greenberg–Hastings Model	51
13.1. Greenberg-Hastings Equation	51
13.2. Explanation	51
14. The Langton’s Ant	52
14.1. The Langton’s Ant Equations	52
14.2. Observations and Phenomena	52
15. The Discrete Parallel Universe of PDE	52
15.1. Motivation	52
15.2. Calculus and Discrete Calculus	53
15.3. ODE and OΔE	54
15.4. Initial and Boundary Conditions	54
15.5. Laplace Transform and Z Transform	55
15.6. Fourier Transform and Discrete Fourier Transform	55
15.7. Discrete and Continuous Green’s Function	56
15.8. Gaussian Distribution and Combinatorics	57
15.9. Discrete and Continuous Dynamical Systems	57
15.10. Manifolds and Networks	58
15.11. Boolean Algebra, Theoretical Computer Science and Number Theory	59
15.12. Summary	59
16. Summary	59
16.1. From Numerical Methods to Language of Complexity	59
17. References	60

Preface

Ever since entering university, I have been trying to model biological evolution. At first, I attempted to use graph theory, but I failed... graph theory could not describe ring species. Then I tried partial differential equations (PDE), but again I failed, as PDE could not describe surface splitting; I wanted to model cellular division, but it was simply beyond reach.

One night, a sudden image flashed in my mind: countless organic molecules colliding near a hydrothermal vent on the seafloor... Ah... this is a dynamical system! I suddenly realized: the initial concentration and distribution of molecules are initial conditions, the size of the environment is the boundary condition, and the temperature, pH, salinity, etc., are non-autonomous terms. This is a dynamical system!

Then I understood that biology is inherently a discrete system. Molecules, cells, individuals—they are all discrete units (discrete space). Reproduction happens across generations (discrete time). So we should not use differential equations. We should use difference equations. But I couldn't figure out what partial difference equations were supposed to be. What do they look like?

Then it suddenly hit me: the Game of Life, the Sandpile Model... I realized these are, in fact, partial difference equations! They all share one essential feature: discrete space and discrete time.

Thus, I inadvertently unified all complex system models. Then I asked myself: partial differential equations describe the evolution of continuous fields. Therefore, partial difference equations describe the evolution of discrete fields!

This led me to systematically replace cellular automata, classical evolutionary theory, and many biological models with a new framework I now call Discrete Field Theory. I began incorporating rich terminology from theoretical physics and classical field theory.

I realized this theory is far too vast for one person to complete alone. Thus, I sincerely invite all scientists to participate in the development and refinement of this framework. Your contributions will be deeply appreciated.

This work is the result of countless nights of failure, reflection, and sudden clarity. I hope this story helps the reader understand not only the theory... but the journey that led to its birth.

Prologue: An Invitation, Not a Declaration

This paper represents only a fragment of a broader and more ambitious framework that I call Discrete Field Theory. I do not claim to have discovered a final theory for complex systems, rather, I am aware that this framework is vast, unfinished, and in need of many more concrete models and rigorous theorems.

As such, I humbly and sincerely invite mathematicians, scientists, and engineers around the world to join in the effort of developing this theory further.

If there are any errors in this work, I welcome constructive criticism and corrections.

If you find value in this paper and wish to contribute or discuss further, please feel free to contact me.

This is my e-mail,
besto2852@gmail.com

1. Introduction to Partial Difference Equations

1.1. Motivation and Background

Many systems in modern biology exhibit inherently discrete organization: molecular interaction networks, cellular assemblies, and ecological populations are composed of countable entities; cells occupy spatially distinct locations; reproduction and generational turnover occur in discrete steps. Even gene expression and mutation often manifest as abrupt, stochastic events rather than smooth, continuous processes.

Nevertheless, the predominant mathematical frameworks for modeling such systems, particularly partial differential equations (PDE) are formulated in terms of continuous variables and differentiable fields [1,2]. While PDE-based models have been highly successful in physics and engineering, their reliance on smoothness and continuity can make them less well-suited for describing discrete interactions, sudden transitions, or spatially distributed logical rules that frequently arise in biological and other complex systems [3,4].

This motivates the question of whether a discrete analogue of PDE, which we term a *Partial Difference Equations* (PΔE), that is, a discrete-space, discrete-time equation governing the evolution of a field defined on a lattice, could serve as a more natural modeling framework for complex, emergent systems. The aim of this work is to develop a functional-analytic foundation for PΔE and to demonstrate their potential for representing systems in which local, discrete interactions give rise to large-scale collective phenomena. We propose that such a formalism offers a mathematically coherent, computationally grounded, and potentially more faithful alternative to continuous PDE in domains where discreteness is an essential feature.

1.2. Discrete Function

In this section, we introduce the concept of a *discrete function*, which serves as the fundamental object in our framework for discrete dynamical systems.

Definition 1 (Discrete Function). A *discrete function* is a mapping

$$f : \Omega \subseteq \mathbb{Z}^n \rightarrow \mathbb{C},$$

where \mathbb{Z}^n is the discrete n -dimensional integer lattice, and \mathbb{C} is the codomain of the function (real, integer, or complex values depending on context).

In this paper, we primarily focus on the case where $f : \mathbb{Z}^n \rightarrow \mathbb{R}$, i.e., real-valued discrete functions.

Example 1 (Discrete Single-variable Function). Let $f : \mathbb{Z} \rightarrow \mathbb{R}$ be defined as

$$f(x) = \sin(x).$$

This is a real-valued function defined on the integer lattice \mathbb{Z} .

Example 2 (Discrete Multivariable Function). Let $u : \mathbb{Z}^2 \rightarrow \mathbb{Z}$ be defined as

$$u(t, x) = \lfloor x + \sin(t) \rfloor.$$

Here, u maps discrete spacetime coordinates (t, x) to an integer value via the floor function applied to a real expression.

1.3. Shift Operator

We define the **shift operator** E^k acting on a discrete function $x(t)$ as

$$E^k x := x(t + k)$$

for any integer $k \in \mathbb{Z}$.

Examples.

$$\begin{aligned}Ex &= x(t+1) \\ E^2x &= x(t+2) \\ E^{-1}x &= x(t-1)\end{aligned}$$

1.4. Partial Shift Operator

Definition. Given a discrete scalar field $u(x_1, x_2, \dots, x_n)$, the *partial shift operator* $E_{x_i}^{k_i}$ acts on u by shifting the i -th coordinate:

$$E_{x_i}^{k_i}u := u(x_1, \dots, x_i + k_i, \dots, x_n)$$

Examples.

$$\begin{aligned}E_t u &= u(t+1, x, y, z) \\ E_x^2 u &= u(t, x+2, y, z)\end{aligned}$$

In this paper, we will write all the Partial Difference Equations in terms of Shift Operators.

1.5. Ordinary Difference Equations

Definition 2 (Ordinary Difference Equation of Order k). An ordinary difference equation of order k is an equation involving a single-variable function $x : \mathbb{Z} \rightarrow \mathbb{C}$, expressed in terms of shift operators:

$$F(E^n x, E^{n-1}x, \dots, Ex, x, E^{-1}x, \dots, E^{-m}x, t) = 0,$$

where $E^i x := x(t+i)$, $n, m \in \mathbb{Z}^+$, $k = m+n$ is the total order, and F is a (possibly nonlinear) function.

Definition 3 (Definition: Linear Ordinary Difference Equation of Order k). A linear ordinary difference equation of order k is an equation of the form:

$$a_n(t) E^n x + a_{n-1}(t) E^{n-1}x + \dots + a_0(t) x + a_{-1}(t) E^{-1}x + \dots + a_{-m}(t) E^{-m}x = g(t),$$

where:

- $x = x(t)$ is the unknown function defined on \mathbb{Z} ,
- $E^j x := x(t+j)$ is the shift operator,
- $a_j(t)$ are known coefficient functions,
- $g(t)$ is the known input or forcing term,
- $m, n \in \mathbb{N}$ and $k = m+n$ is the total order.

1.6. Partial Difference Equation

Definition 4 (Partial Difference Equation). Let $u : \mathbb{Z}^n \rightarrow \mathbb{R}$ (or \mathbb{C}) be a scalar function defined on the discrete lattice. A *partial difference equation (PΔE) with constant order* is an equation of the form:

$$F\left(\left\{E_{x_1}^{k_1} E_{x_2}^{k_2} \dots E_{x_n}^{k_n} u\right\}_{(k_1, k_2, \dots, k_n) \in S}, x_1, x_2, \dots, x_n\right) = 0,$$

where:

- $u = u(x_1, x_2, \dots, x_n)$

- F is a given function, which may be linear or nonlinear;
- E_{x_i} denotes the shift operator in the x_i direction, defined by

$$E_{x_i}^{k_i} u = u(x_1, x_2, \dots, x_i + k_i, \dots, x_n);$$

- $S \subset \mathbb{Z}^n$ is a finite index set determining the set of applied shifts.

1.7. The Order of a Partial Difference Equation

We propose two types of order to classify the structure of partial difference equations with constant order: **structural order** and **absolute order**.

Definition 5 (Structural Order of Partial Difference Equation). Let $u = u(x_1, x_2, \dots, x_n)$ be a scalar function defined on a discrete lattice. A partial difference equation is of the form

$$F\left(\left\{E_{x_1}^{k_1} E_{x_2}^{k_2} \cdots E_{x_n}^{k_n} u\right\}_{(k_1, k_2, \dots, k_n) \in S}, x_1, x_2, \dots, x_n\right) = 0,$$

where $S \subset \mathbb{Z}^n$ is a finite index set of shifts.

For each variable x_i , define

$$\text{ord}(x_i) = \max(k_i) - \min(k_i),$$

where $\max(k_i)$ is the maximum order of the shift operator in x_i direction, $\min(k_i)$ is the minimum order of the shift operator in x_i direction, and $\text{ord}(x_i)$ is the structural order in x_i direction.

Then, the **structural order** of the equation is defined as

$$\max(\text{ord}(x_1), \text{ord}(x_2), \dots, \text{ord}(x_n)).$$

Definition 6 (Absolute Order of Partial Difference Equation). Let $u = u(x_1, x_2, \dots, x_n)$ and consider a partial difference equation of the form

$$F\left(\left\{E_{x_1}^{k_1} E_{x_2}^{k_2} \cdots E_{x_n}^{k_n} u\right\}_{(k_1, k_2, \dots, k_n) \in S}, x_1, x_2, \dots, x_n\right) = 0,$$

where $S \subset \mathbb{Z}^n$ is a finite index set.

If for all $(k_1, \dots, k_n) \in S$, the sum

$$|k_1| + |k_2| + \cdots + |k_n| \leq k,$$

then the equation is said to have **absolute order** k .

In addition:

- A shift operator E_x^n is said to be of order $|n|$.
- A mixed shift operator $E_{x_1}^{k_1} E_{x_2}^{k_2} \cdots E_{x_n}^{k_n}$ is said to be of order $|k_1| + |k_2| + \cdots + |k_n|$.

1.8. Comparison with Ordinary Difference Equations

The essential difference lies in the function domain and structure:

- **Ordinary Difference Equation (ODE):** Involves a single-variable function $x : \mathbb{Z} \rightarrow \mathbb{C}$, typically evolving along a one-dimensional time axis.
- **Partial Difference Equation (PDE):** Involves a multi-variable function $u : \mathbb{Z}^n \rightarrow \mathbb{C}$, evolving over a multi-dimensional discrete lattice.
- PDE equations utilize *partial shift operators*, such as E_{x_i} , acting independently on each coordinate axis, introducing more geometric and structural complexity.

1.9. Comparison with Partial Differential Equations

Partial difference equations can also be viewed as the discrete analogues of partial differential equations. Key distinctions include:

- **Partial Differential Equations (PDE):** Defined over continuous space-time domains. Dynamics are governed by partial derivatives, such as $\frac{\partial u}{\partial x}$ or $\frac{\partial^2 u}{\partial t^2}$.
- **Partial Difference Equations (PΔE):** Defined over discrete lattice domains, where partial derivatives are replaced by *partial difference operators* or *partial shift operators*, e.g. $E_{x_i} u = u(\dots, x_i + 1, \dots)$.
- PΔE capture the dynamics of discrete systems such as cellular automata, lattice models, or numerical schemes, and often exhibit different phenomena such as intrinsic stochasticity or discrete chaos.

1.10. Advantages of Rewriting Complex Systems as Partial Difference Equations

1. **Elegant and Unified Framework**

Partial Difference Equations provide a universal and elegant mathematical language to describe a wide range of complex systems. Instead of relying on a fragmented set of ad hoc models, many seemingly unrelated systems can be encoded using the same formalism. This promotes unification across disciplines such as biology, physics, and computer science.

2. **Mathematical Rigor and Analytical Tools**

By expressing complex systems through Partial Difference Equations, we enable the use of powerful tools from modern mathematics, including functional analysis, topology, and chaos theory. This increases the precision, clarity, and theoretical strength of the analysis.

3. **Extending Modeling Paradigms**

Many classical models of complex systems such as Conway's Game of Life or the sandpile model, are *static* in the sense that individual entities do not move, and they typically involve only a single scalar field. Reformulating these systems as Partial Difference Equations makes it straightforward to introduce *multiple field coupling*, or even *discrete vector fields* to represent additional attributes of the entities, such as phenotypic traits or movement directions. This greatly expands the range of phenomena that can be modeled within the same theoretical framework.

1.11. Scope and Limitations

In this paper, we first introduce the definition of Partial Difference Equations (PΔE) and provide a brief overview of the linear case. We then present the concepts of discrete function spaces, discrete operators, and the discrete Green's function. Several well-known discrete models, such as cellular automata, the Abelian sandpile model, and the Ising model, will be reformulated within the PΔE framework, though without an in-depth study of each individual model.

Finally, we compare Partial Difference Equations with their continuous counterparts, Partial Differential Equations (PDE). While PDE naturally involve tools from calculus, differential geometry, and probability theory (e.g., Gaussian distributions), PΔE instead draw upon discrete calculus, combinatorics, discrete geometry, and number theory. Due to space limitations, topics such as explicit solution methods for linear equations and regularity theory of solutions are beyond the scope of this work.

Future Work.

Future investigations will address topics deliberately omitted here, such as analytical solution methods for linear PΔE, the regularity theory of solutions, and deeper exploration of specific discrete models. In particular, extending the PΔE framework to incorporate stochastic effects, irregular lattice geometries, and more general coupling topologies represents a promising direction for further research.

1.12. Classification and Examples

Partial Difference Equations (PΔE) can be categorized along several axes, including linearity, time-dependence, and dimensionality. Below are some common types:

1. **Linear PΔE:** A PΔE is said to be linear if it is linear in u and all its shifted terms. For example:

$$E_t u = \frac{1}{2}(E_x^{-1} u + E_x u)$$

This equation represents a discrete diffusion process, where the future state is the average of neighbouring spatial values.

2. **Nonlinear PΔE:** Nonlinear PΔE contain nonlinear terms involving u or its shifts. For example:

$$E_t u = u + \sin(E_x^{-1} u - E_x u)$$

Such systems can exhibit complex behaviour, including chaos and pattern formation.

3. **Autonomous PΔE:** If the evolution rule is independent of the explicit time step t , the equation is said to be autonomous:

$$E_t u = F(\{E_x^k u\}_{k \in \mathbb{N}})$$

These are time-invariant systems and useful for studying equilibrium dynamics.

4. **Non-Autonomous PΔE:** If the update rule explicitly depends on t or x , the system is non-autonomous:

$$E_t u = F(\{E_x^k u\}_{k \in \mathbb{N}}, t, x)$$

These equations are useful for modeling time-varying or spatially heterogeneous environments.

5. **System of PΔE:** A coupled system of multiple interacting fields:

$$\begin{cases} E_t u = f(u, E_x u, E_x^{-1} u, v, E_x v, E_x^{-1} v) \\ E_t v = g(u, E_x u, E_x^{-1} u, v, E_x v, E_x^{-1} v) \end{cases}$$

Such systems arise in biology, physics, and complex networks.

1.13. Notations

Definition 7 (Difference Operator). Let $x = x(t)$. The forward difference operator is defined as

$$\Delta x := x(t+1) - x(t)$$

Definition 8 (Partial Difference Operator). Let $u = u(t, x, y, z)$. The partial difference operator in the x -direction is defined as

$$\Delta_x u := u(t, x+1, y, z) - u(t, x, y, z)$$

Definition 9 (Shift Operator). Let $x = x(t)$. The shift operator is defined as

$$E x := x(t+1)$$

Definition 10 (Partial Shift Operator). Let $u = u(t, x, y, z)$. The partial shift operator is defined as

$$E_x u := u(t, x+1, y, z)$$

Definition 11 (Central Difference). Let $x = x(t)$. The central difference is defined as

$$\Delta^c x := \frac{x(t+1) - x(t-1)}{2}$$

Definition 12 (Partial Central Difference). Let $u = u(t, x)$. The partial central difference is defined as

$$\Delta_x^c u := \frac{u(t, x+1) - u(t, x-1)}{2}$$

Definition 13 (Backward Difference Operator). Let $x = x(t)$. The backward difference operator is defined as

$$\delta x := x(t) - x(t-1)$$

Definition 14 (Partial Backward Difference Operator). Let $u = u(t, x)$. The partial backward difference operator is defined as

$$\delta_x u := u(t, x) - u(t, x-1)$$

Definition 15 (Discrete Gradient). The discrete gradient is defined as

$$\nabla := (\Delta_x, \Delta_y, \Delta_z)$$

Definition 16 (Backward Gradient). The backward gradient is defined as

$$\nabla_\delta := (\delta_x, \delta_y, \delta_z)$$

Definition 17 (Central Gradient). The central gradient is defined as

$$\nabla_c := (\Delta_x^c, \Delta_y^c, \Delta_z^c)$$

Definition 18 (Discrete Laplacian). The discrete Laplacian is defined as

$$\nabla^2 u := \nabla_\delta \cdot \nabla u = \delta_x \Delta_x u + \delta_y \Delta_y u + \delta_z \Delta_z u$$

Definition 19 (von Neumann neighbourhood). The von Neumann neighbourhood V is defined as the set of lattice displacements:

$$V = \{(0, 1), (0, -1), (1, 0), (-1, 0)\}$$

Definition 20 (Moore neighbourhood). The Moore neighbourhood M is defined as the set of lattice displacements:

$$M = \{(0, 1), (0, -1), (1, 0), (-1, 0), (1, 1), (1, -1), (-1, 1), (-1, -1)\}$$

Definition 21 (Moore Laplacian (in 2D)). The Moore Laplacian is defined as

$$\nabla_M^2 u := \left(\sum_{(i,j) \in M} u(t, x+i, y+j) \right) - 8u(t, x, y)$$

where M is the Moore neighbourhood excluding the center point.

Definition 22 (Kronecker Delta Function). The Kronecker delta function $\delta(x)$ is defined as:

$$\delta(x) = \begin{cases} 1, & \text{if } x = 0 \\ 0, & \text{otherwise} \end{cases}$$

Definition 23 (Heaviside Step Function). The Heaviside step function $\theta(x)$ is defined as:

$$\theta(x) = \begin{cases} 1, & \text{if } x \geq 0 \\ 0, & \text{otherwise} \end{cases}$$

Definition 24 (Signum Function). The signum function, denoted by $\text{sign}(x)$, is defined as:

$$\text{sign}(x) = \begin{cases} +1 & \text{if } x > 0, \\ 0 & \text{if } x = 0, \\ -1 & \text{if } x < 0. \end{cases}$$

Definition 25 (Modulo Function). The modulo function $\text{mod}_n(x)$ is defined as:

$$\text{mod}_n(x) = x - n \left\lfloor \frac{x}{n} \right\rfloor$$

where $\lfloor \cdot \rfloor$ denotes the floor function, and $n \in \mathbb{R}$.

2. Linear Partial Difference Equations

2.1. Motivation

Although most well-known cellular automata are nonlinear partial difference equations, we believe it is inappropriate to skip the study of linear partial difference equations and proceed directly to the nonlinear case. However, due to limitations in our current capabilities, this section will be limited to introducing the formal definition of linear partial difference equations, along with three classical examples: the discrete one-dimensional transport equation, the discrete one-dimensional diffusion equation, and the discrete one-dimensional wave equation.

We will present a few particular solutions for each equation, rather than attempting to develop a general solution method.

2.2. Definition

Definition 26 (Linear Partial Difference Equation). Let $u = u(x_1, x_2, \dots, x_n)$ be a scalar function defined on the discrete lattice \mathbb{Z}^n . A **linear partial difference equation** is an equation of the form

$$\sum_{(k_1, \dots, k_n) \in S} a_{k_1 k_2 \dots k_n}(x_1, x_2, \dots, x_n) E_{x_1}^{k_1} E_{x_2}^{k_2} \dots E_{x_n}^{k_n} u(x_1, x_2, \dots, x_n) = f(x_1, x_2, \dots, x_n),$$

where:

- $S \subset \mathbb{Z}^n$ is a finite index set,
- E_{x_i} is the shift operator in the x_i direction: $E_{x_i} u(\dots, x_i, \dots) = u(\dots, x_i + 1, \dots)$,
- $a_{k_1 k_2 \dots k_n}$ and f are known functions on \mathbb{Z}^n .

Definition 27 (Linear Partial Difference Equation of Absolute Order k). Let $u = u(x_1, x_2, \dots, x_n)$ be a scalar function on a discrete lattice \mathbb{Z}^n . A **linear partial difference equation of absolute order k** is an equation of the form:

$$\sum_{|k_1| + \dots + |k_n| \leq k} a_{k_1 k_2 \dots k_n}(x_1, x_2, \dots, x_n) E_{x_1}^{k_1} E_{x_2}^{k_2} \dots E_{x_n}^{k_n} u = f(x_1, x_2, \dots, x_n),$$

- $(k_1, k_2, \dots, k_n) \in \mathbb{Z}^n$.
- E_{x_i} is the shift operator in the x_i direction: $E_{x_i} u(\dots, x_i, \dots) = u(\dots, x_i + 1, \dots)$,
- $a_{k_1 k_2 \dots k_n}$ and f are known functions on \mathbb{Z}^n .

2.3. Discrete 1D Transport Equation

We consider the Discrete Transport Equation:

$$E_t u = E_x^k u$$

where $u : \mathbb{Z}^2 \rightarrow \mathbb{R}$ is defined on discrete spacetime.

Initial condition:

$$u(0, x) = f(x)$$

We impose no boundary condition.

Interpretation:

Each value is transported rigidly along a discrete path. The effective velocity is $-k$, since data moves from $x + k$ to x .

Characteristic curve:

$$x(t) = x_0 - kt$$

Solution

We consider the partial difference equation

$$E_t u = E_x^k u, \quad k \in \mathbb{Z},$$

with no boundary conditions.

Let us define a new variable:

$$\xi = x + kt,$$

and define a new function

$$v(t, \xi) := u(t, x).$$

Then we compute the time-shift of v :

$$E_t v = v(t+1, \xi) = v(t+1, x + k(t+1)) = v(t+1, x + kt + k) = v(t+1, \xi + k) = E_\xi^k v = E_t E_x^k u$$

$$E_t v = E_t E_x^k u$$

$$v = E_x^k u$$

On the other hand, from the original equation:

$$E_t u = u(t+1, x) = E_x^k u(t, x) = u(t, x + k).$$

Substitute $u(t, x) = v(t, \xi)$, we get:

$$E_t u = u(t+1, x) = v(t+1, \xi) = E_t v.$$

And

$$E_x^k u = v$$

Thus,

$$E_t v = v,$$

or equivalently,

$$E_t v = v.$$

This is just an Ordinary Difference Equation.

Now we define a new function of ξ only:

$$v(t, \xi) = f(\xi),$$

which solves the equation since

$$E_t v = v.$$

Therefore, the general solution is

$$v(t, \xi) = f(\xi) = f(x + kt),$$

so that

$$u(t, x) = f(x + kt),$$

where $f(x) = u(0, x)$ is the initial condition.

Conclusion: The solution of the partial difference equation

$$E_t u = E_x^k u$$

is given by

$$u(t, x) = f(x + kt),$$

where f is the initial profile of u at time $t = 0$.

A General Form of Discrete 1D Linear Transport Equation

$$E_t u = p(t, x) E_x^{k(t, x)} u + q(t, x) u + g(t, x)$$

The operator $E_x^{k(t, x)}$ is linear because it does not depend on the function u . Although this may seem counterintuitive at first, we can rigorously verify its linearity:

$$E_x^{k(t, x)}(au + bv) = a u(t, x + k(t, x)) + b v(t, x + k(t, x)) = a E_x^{k(t, x)} u + b E_x^{k(t, x)} v$$

Thus, $E_x^{k(t, x)}$ is a linear operator.

Furthermore, we observe that the order of a difference operator does not necessarily need to be a constant; it can also be a function. To avoid confusion, we refer to the classical case where the order is constant as **difference equations with constant order**. When the order is a function, we refer to them as **difference equations with variable order**. This is a type of **Functional Partial Difference Equations**. In this paper, we will only discuss about Partial Difference Equations with constant order.

Unless otherwise specified, the term "difference equation" typically assumes constant order by default.

2.4. Discrete 1D Diffusion Equation

In this section, we propose a discrete analogue of the classical one-dimensional diffusion equation. The equation is given by

$$\Delta_t u = \alpha \nabla^2 u,$$

where

$$u = u(t, x), \quad u : \mathbb{Z}^2 \rightarrow \mathbb{R},$$

$$\Delta_t u := E_t u - u,$$

$$\nabla^2 u := E_x u + E_x^{-1} u - 2u,$$

and $\alpha \in \mathbb{R}$ is a constant representing the diffusion coefficient.

Physical Interpretation.

This equation models the diffusion of some quantity (e.g., heat, particles, or chemical concentration) across a one-dimensional discrete lattice over discrete time steps. The Laplacian term $\nabla^2 u$ captures the spatial flow of the quantity due to differences between neighbouring sites, while $\Delta_t u$ describes the temporal evolution of the system.

The Discrete 1D Diffusion Equation models symmetric spreading on a 1D lattice.

Special Case

We consider the case where the diffusion coefficient $\alpha = \frac{1}{2}$. Then the equation becomes the discrete heat equation:

$$E_t u = \frac{1}{2} (E_x u + E_x^{-1} u).$$

We first consider the initial condition:

$$u(0, x) = \delta(x),$$

with no boundary condition.

Then the solution is given by the discrete heat kernel:

$$u(t, x) = \delta(\text{mod}_2(x+t)) C\left(t, \frac{x+t}{2}\right) \left(\frac{1}{2}\right)^t,$$

where the binomial coefficient $C(x, y)$ is defined as:

$$C(x, y) = \begin{cases} \frac{x!}{(x-y)!y!}, & \text{if } 0 \leq y \leq x, \\ 0, & \text{otherwise,} \end{cases} \quad \text{with } C : \mathbb{Z}^2 \rightarrow \mathbb{Z}.$$

We omit the proof since this solution is well known as the discrete heat kernel, or Green's function for the discrete heat equation.

If the initial condition is instead:

$$u(0, x) = f(x),$$

with no boundary condition, then the solution is given by:

$$u(t, x) = \sum_{s \in \mathbb{Z}} f(s) \delta(\text{mod}_2(x-s+t)) C\left(t, \frac{x-s+t}{2}\right) \left(\frac{1}{2}\right)^t.$$

Here, $\text{mod}_2(x) := x - 2 \lfloor \frac{x}{2} \rfloor$ is the modulo function.

2.5. Discrete 1D Wave Equation

We define the discrete 1D wave equation as

$$\delta_t \Delta_t u = c^2 \nabla^2 u$$

where $u = u(t, x)$, with $u : \mathbb{Z}^2 \rightarrow \mathbb{R}$, and the operators are defined via shift operators:

$$\begin{aligned} \delta_t \Delta_t u &= E_t u + E_t^{-1} u - 2u, \\ \delta_t u &= u - E_t^{-1} u \quad (\text{backward difference}), \\ \nabla^2 u &= E_x u + E_x^{-1} u - 2u \quad (\text{discrete Laplacian}). \end{aligned}$$

Physical Interpretation. This equation models a system of coupled oscillators arranged on a one-dimensional lattice. Each oscillator interacts with its nearest neighbours, and the term $\nabla^2 u$ represents the net restoring force. The parameter c is a constant that governs the wave propagation speed in the discrete medium.

Special Case

We consider the case where $c = 1$. Then the equation becomes:

$$E_t u = E_x^{-1} u + E_x u - E_t^{-1} u$$

with initial conditions:

$$u(0, x) = \delta(x), \quad u(1, x) = \delta(x)$$

and no boundary condition (i.e., defined on the full lattice \mathbb{Z}).

Exact Solution:

$$u(t, x) = \delta(t)\delta(x) + \theta(t-1)(-1)^{|x|+t-1} \cdot \text{com}(x, t-1)$$

where $\text{com}(x, t)$ is a compact support indicator function defined by:

$$\text{com}(x, a) := \begin{cases} 1, & \text{if } |x| \leq a \\ 0, & \text{otherwise} \end{cases}$$

This can be generalized to be centered at an arbitrary point x_0 :

$$\text{com}(x - x_0, a) := \begin{cases} 1, & \text{if } |x - x_0| \leq a \\ 0, & \text{otherwise} \end{cases}$$

The alternating sign $(-1)^{|x|+t-1}$ introduces a checkerboard-like structure in the wave propagation, while the compact support ensures finite spread with speed 1.

The following picture is the spatiotemporal plot of the Discrete 1D Wave Equation.

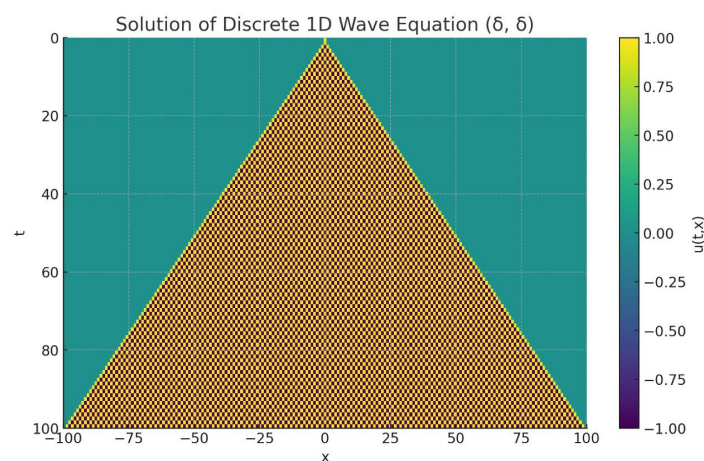


Figure 1. Spatiotemporal plot of the 1D Wave Equation

3. Discrete Function Spaces and Operators

3.1. Motivation

This section lays the theoretical foundation for the analysis of partial difference equations (PΔE). We systematically introduce the structure and properties of discrete function spaces, difference and

shift operators, discrete analogues of Hilbert spaces, the Kronecker delta function, discrete Green's functions, and boundary value problems.

The notation and operator definitions adopted here largely follow established conventions in the literature (see, e.g., [5]). The formulation and conceptual framework are inspired by the classical theory of partial differential equations (see, e.g., [6]), with appropriate adaptations to the discrete setting.

This section provides a rigorous functional-analytic formulation of shift operators. Readers primarily interested in applications may proceed directly to Section 4.

3.2. Discrete Function Spaces

Definition 28 (Discrete Function Space). Let $\Omega \subseteq \mathbb{Z}^n$ be a discrete domain. We define the **discrete function space** over Ω as

$$F(\Omega) := \{f : \Omega \rightarrow \mathbb{C}\},$$

i.e., the set of all functions from Ω to the complex numbers.

Definition 29 (Discrete \mathcal{L}^p Space). Let $\Omega \subseteq \mathbb{Z}^n$. We define the discrete \mathcal{L}^p space for $1 \leq p < \infty$ as

$$\mathcal{L}^p(\Omega) := \left\{ f : \Omega \rightarrow \mathbb{C} \left| \sum_{\mathbf{x} \in \Omega} |f(\mathbf{x})|^p < \infty \right. \right\},$$

with the corresponding \mathcal{L}^p norm defined by

$$\|f\|_p := \left(\sum_{\mathbf{x} \in \Omega} |f(\mathbf{x})|^p \right)^{1/p}.$$

For $p = \infty$, we define

$$\mathcal{L}^\infty(\Omega) := \left\{ f : \Omega \rightarrow \mathbb{C} \left| \sup_{\mathbf{x} \in \Omega} |f(\mathbf{x})| < \infty \right. \right\},$$

with the corresponding \mathcal{L}^∞ norm given by

$$\|f\|_\infty := \sup_{\mathbf{x} \in \Omega} |f(\mathbf{x})|.$$

Definition 30 (Discrete Hilbert Space). Let $\Omega \subseteq \mathbb{Z}^n$. The discrete Hilbert space is the space $\mathcal{L}^2(\Omega)$ defined as

$$\mathcal{L}^2(\Omega) := \left\{ f : \Omega \rightarrow \mathbb{C} \left| \sum_{\mathbf{x} \in \Omega} |f(\mathbf{x})|^2 < \infty \right. \right\},$$

equipped with the inner product

$$\langle f, g \rangle := \sum_{\mathbf{x} \in \Omega} f(\mathbf{x}) \overline{g(\mathbf{x})}, \quad \text{for all } f, g \in \mathcal{L}^2(\Omega).$$

This inner product satisfies the following properties for all $f, g, h \in \mathcal{L}^2(\Omega)$ and all scalars $\alpha \in \mathbb{C}$:

1. **Conjugate Symmetry:**

$$\langle f, g \rangle = \overline{\langle g, f \rangle}.$$

2. **Linearity in the First Argument:**

$$\langle \alpha f + h, g \rangle = \alpha \langle f, g \rangle + \langle h, g \rangle.$$

3. **Positive Definiteness:**

$$\langle f, f \rangle \geq 0, \quad \text{and} \quad \langle f, f \rangle = 0 \iff f = 0.$$

The norm induced by this inner product is

$$\|f\|_2 := \sqrt{\langle f, f \rangle} = \left(\sum_{\mathbf{x} \in \Omega} |f(\mathbf{x})|^2 \right)^{1/2}.$$

Moreover, $\mathcal{L}^2(\Omega)$ is complete with respect to this norm, and hence forms a Hilbert space.

3.3. Operators

Definition 31 (Discrete Operator). Let V, W be discrete function spaces, i.e., subspaces of $F(\Omega)$, where $\Omega \subseteq \mathbb{Z}^n$.

A **discrete operator** is a map

$$T : V \rightarrow W,$$

which assigns to each function $f \in V$ a function $Tf \in W$.

Definition 32 (Shift Operator). Let $x : \mathbb{Z} \rightarrow \mathbb{C}$. We define the **shift operator** E^k acting on a discrete function $x(t)$ by

$$E^k x := x(t + k)$$

for any integer $k \in \mathbb{Z}$.

Definition 33 (Partial Shift Operator). Given a discrete scalar field $u(x_1, x_2, \dots, x_n)$, the **partial shift operator** $E_{x_i}^{k_i}$ acts on u by shifting the i -th coordinate:

$$E_{x_i}^{k_i} u := u(x_1, \dots, x_i + k_i, \dots, x_n)$$

for any integer $k_i \in \mathbb{Z}$.

Definition 34 (Difference Operator). Let $x : \mathbb{Z} \rightarrow \mathbb{C}$. We define the (forward) **difference operator** Δ by

$$\Delta x := x(t + 1) - x(t)$$

We may define the shift operator E as $Ex := x(t + 1)$, so that the difference operator can be written compactly as

$$\Delta x = Ex - x$$

Definition 35 (Partial Difference Operator). Let $u : \mathbb{Z}^n \rightarrow \mathbb{C}$, and denote $u = u(x_1, x_2, \dots, x_n)$. The **partial difference operator** with respect to the variable x_i is defined as:

$$\Delta_{x_i} u := E_{x_i} u - u$$

where E_{x_i} is the partial shift operator acting on the i -th coordinate:

$$E_{x_i} u := u(x_1, \dots, x_i + 1, \dots, x_n)$$

Definition 36 (Backward Difference Operator). Let $x : \mathbb{Z} \rightarrow \mathbb{C}$ be a discrete function with $x = x(t)$. Then the **backward difference operator** is defined as

$$\delta x := x(t) - x(t - 1) = x - E^{-1}x$$

Definition 37 (Partial Backward Difference Operator). Let

$$u : \mathbb{Z}^n \rightarrow \mathbb{C}, \quad u = u(x_1, x_2, \dots, x_n)$$

Then the **partial backward difference operator** with respect to x_i is defined as

$$\delta_{x_i} u := u - E_{x_i}^{-1} u$$

where $E_{x_i}^{-1}$ is the backward shift operator acting on the x_i -th coordinate.

Definition 38 (Central Difference Operator). Let $x : \mathbb{Z} \rightarrow \mathbb{C}$, with $x = x(t)$. Then the **central difference operator** is defined as

$$\Delta^c x := \frac{1}{2} (Ex - E^{-1}x)$$

where E is the forward shift operator: $Ex = x(t+1)$, and $E^{-1}x = x(t-1)$.

Definition 39 (Partial Central Difference Operator). Let $u : \mathbb{Z}^n \rightarrow \mathbb{C}$, where $u = u(x_1, x_2, \dots, x_n)$. Then the **partial central difference operator** with respect to x_i is defined as

$$\delta_{x_i} u := \frac{1}{2} (E_{x_i} u - E_{x_i}^{-1} u),$$

where E_{x_i} is the shift operator in the x_i -direction, defined by

$$E_{x_i} u = u(x_1, \dots, x_i + 1, \dots, x_n).$$

Definition 40 (Discrete Gradient). Let $u : \mathbb{Z}^n \rightarrow \mathbb{C}$ be a discrete scalar function, where $u = u(x_1, x_2, \dots, x_n)$. Then the **discrete gradient** of u is defined as the vector:

$$\nabla u := (\Delta_{x_1} u, \Delta_{x_2} u, \dots, \Delta_{x_n} u)$$

where Δ_{x_i} is the forward partial difference operator defined by

$$\Delta_{x_i} u := u(x_1, \dots, x_i + 1, \dots, x_n) - u(x_1, \dots, x_i, \dots, x_n)$$

for each $i = 1, 2, \dots, n$.

Definition 41 (Backward Gradient). Let

$$u : \mathbb{Z}^n \rightarrow \mathbb{C}, \quad u = u(x_1, x_2, \dots, x_n),$$

then the **backward gradient** is defined as

$$\nabla_{\delta} u = (\delta_{x_1} u, \delta_{x_2} u, \dots, \delta_{x_n} u),$$

where each $\delta_{x_i} u$ denotes the **partial backward difference** in the x_i -direction.

Definition 42 (Central Gradient). Let

$$u : \mathbb{Z}^n \rightarrow \mathbb{C}, \quad u = u(x_1, x_2, \dots, x_n),$$

be a discrete function on the n -dimensional integer lattice. Then the **Central Gradient** is defined as the vector

$$\nabla_c u := (\Delta_{x_1}^c u, \Delta_{x_2}^c u, \dots, \Delta_{x_n}^c u),$$

where each $\Delta_{x_i}^c u$ denotes the partial central difference of u with respect to the variable x_i .

Definition 43 (Discrete Laplacian). Let

$$u : \mathbb{Z}^n \rightarrow \mathbb{C}, \quad u = u(x_1, x_2, \dots, x_n),$$

then the **Discrete Laplacian** is defined as

$$\nabla^2 u = \nabla_\delta \cdot \nabla u = \sum_{i=1}^n \delta_{x_i} \Delta_{x_i} u,$$

where $\nabla u = (\Delta_{x_1} u, \dots, \Delta_{x_n} u)$ is the **discrete gradient**, and $\nabla_\delta = (\delta_{x_1}, \dots, \delta_{x_n})$ is the **backward gradient**.

Definition 44 (Linear Operator). Let $\Omega \subseteq \mathbb{Z}^n$, and let

$$T : \mathcal{D}(T) \subseteq \mathcal{L}^2(\Omega) \rightarrow \mathcal{L}^2(\Omega)$$

be a map with domain $\mathcal{D}(T)$. We say that T is a **linear operator** if for all $u, v \in \mathcal{D}(T)$ and all scalars $\alpha, \beta \in \mathbb{C}$, the following holds:

$$T(\alpha u + \beta v) = \alpha T(u) + \beta T(v).$$

Definition 45 (Linear Shift Operator). Let $\Omega \subseteq \mathbb{Z}^n$ and $m \in \mathbb{N}$. A **linear shift operator** L of absolute order at most m is any map of the form

$$L(u(\mathbf{x})) = \sum_{\|\alpha\|_1 \leq m} a^\alpha(\mathbf{x}) E^\alpha u(\mathbf{x}), \quad \mathbf{x} \in \Omega,$$

where

$$\|\alpha\|_1 := |\alpha_1| + \dots + |\alpha_n|, \quad E^\alpha u(\mathbf{x}) := u(\mathbf{x} + \alpha),$$

and the coefficient functions $a^\alpha : \Omega \rightarrow \mathbb{C}$ are given. Here $E_{x_i}^k$ denotes the one-step shift in the i -th coordinate,

$$E_{x_i}^k u(x_1, \dots, x_i, \dots, x_n) := u(x_1, \dots, x_i + k, \dots, x_n),$$

so that $E^\alpha = E_{x_1}^{\alpha_1} \dots E_{x_n}^{\alpha_n}$, where $\alpha = (\alpha_1, \alpha_2, \dots, \alpha_n) \in \mathbb{Z}^n$. We take $\mathcal{D}(L)$ to be the set of all $u \in \mathcal{L}^2(\Omega)$ for which the right-hand side is well-defined and belongs to $\mathcal{L}^2(\Omega)$.

Proposition 1. Let L be a linear shift operator on a function space $F(\Omega)$, where $\Omega \subseteq \mathbb{Z}^n$. Then the solution set of the homogeneous partial difference equation

$$L(u) = 0$$

is a function subspace (which is also a vector space) of $F(\Omega)$.

Proof. If $u, v \in \ker L$ and $\alpha, \beta \in \mathbb{C}$, then

$$L(\alpha u + \beta v) = \alpha L(u) + \beta L(v) = 0,$$

so $\alpha u + \beta v \in \ker L$. \square

Definition 46 (Adjoint Operator). Let $\mathcal{L}^2(\Omega)$ be a discrete Hilbert space with inner product

$$\langle f, g \rangle := \sum_{\mathbf{x} \in \Omega} f(\mathbf{x}) \overline{g(\mathbf{x})}.$$

Let $T : \mathcal{D}(T) \subseteq \mathcal{L}^2(\Omega) \rightarrow \mathcal{L}^2(\Omega)$ be a linear operator.

We say that a linear operator $T^* : \mathcal{D}(T^*) \subseteq \mathcal{L}^2(\Omega) \rightarrow \mathcal{L}^2(\Omega)$ is the **adjoint** of T if:

$$\langle Tf, g \rangle = \langle f, T^*g \rangle \quad \text{for all } f \in \mathcal{D}(T), g \in \mathcal{D}(T^*).$$

Here, $\mathcal{D}(T^*)$ consists of all $g \in \mathcal{L}^2(\Omega)$ such that the map

$$f \mapsto \langle Tf, g \rangle$$

is continuous (i.e., bounded) on $\mathcal{D}(T)$.

Example 3. Let $\Omega \subseteq \mathbb{Z}^n$ be shift-invariant under $\alpha \in \mathbb{Z}^n$ (e.g. $\Omega = \mathbb{Z}^n$, or Ω with periodic boundary conditions), and consider the Hilbert space $\mathcal{L}^2(\Omega)$ with inner product

$$\langle f, g \rangle := \sum_{\mathbf{x} \in \Omega} f(\mathbf{x}) \overline{g(\mathbf{x})}.$$

For $\alpha \in \mathbb{Z}^n$, define the (multi-dimensional) shift

$$(E^\alpha u)(\mathbf{x}) := u(\mathbf{x} + \alpha),$$

where

$$E^\alpha = E_{x_1}^{\alpha_1} \cdots E_{x_n}^{\alpha_n}, \quad \alpha = (\alpha_1, \dots, \alpha_n) \in \mathbb{Z}^n$$

is the vector representing the direction (and magnitude) of the shift.

Then for all $u, v \in \mathcal{L}^2(\Omega)$,

$$\begin{aligned} \langle E^\alpha u, v \rangle &= \sum_{\mathbf{x} \in \Omega} u(\mathbf{x} + \alpha) \overline{v(\mathbf{x})} = \sum_{\mathbf{y} \in \Omega} u(\mathbf{y}) \overline{v(\mathbf{y} - \alpha)} \\ &= \langle u, E^{-\alpha} v \rangle. \end{aligned}$$

Hence the adjoint of the shift is the opposite shift:

$$(E^\alpha)^* = E^{-\alpha}.$$

Definition 47 (Unitary Operator). Let $\Omega \subseteq \mathbb{Z}^n$ and let

$$U : \mathcal{D}(U) \subseteq \mathcal{L}^2(\Omega) \rightarrow \mathcal{L}^2(\Omega)$$

be a linear operator with domain $\mathcal{D}(U)$. We say that U is **unitary** if:

$$U^*U = UU^* = I,$$

that is,

$$U^* = U^{-1},$$

where U^* is the adjoint of U and U^{-1} is the inverse operator of U . Equivalently, U is unitary if it preserves the inner product:

$$\langle Uf, Ug \rangle = \langle f, g \rangle \quad \text{for all } f, g \in \mathcal{L}^2(\Omega).$$

Proposition 2. Let $\Omega \subseteq \mathbb{Z}^n$ be shift-invariant under $\pm\alpha \in \mathbb{Z}^n$ (e.g. $\Omega = \mathbb{Z}^n$ or Ω with periodic boundary conditions). Consider the Hilbert space $\mathcal{L}^2(\Omega)$ with inner product

$$\langle f, g \rangle := \sum_{\mathbf{x} \in \Omega} f(\mathbf{x}) \overline{g(\mathbf{x})}.$$

For $\alpha \in \mathbb{Z}^n$, define the shift operator

$$(E^\alpha u)(\mathbf{x}) := u(\mathbf{x} + \alpha).$$

Then E^α is a unitary operator on $\mathcal{L}^2(\Omega)$.

Proof. From the calculation in the previous example, we have

$$\langle E^\alpha u, v \rangle = \langle u, E^{-\alpha} v \rangle,$$

which shows $(E^\alpha)^* = E^{-\alpha}$. Since $E^\alpha E^{-\alpha} = E^{-\alpha} E^\alpha = I$, it follows that

$$(E^\alpha)^* = (E^\alpha)^{-1}.$$

Thus E^α is unitary by definition. Equivalently, for all $u \in \mathcal{L}^2(\Omega)$,

$$\|E^\alpha u\|_2^2 = \sum_{\mathbf{x} \in \Omega} |u(\mathbf{x} + \alpha)|^2 = \sum_{\mathbf{y} \in \Omega} |u(\mathbf{y})|^2 = \|u\|_2^2,$$

so E^α preserves the norm and is surjective. \square

Definition 48 (Self-adjoint Operator). Let $\Omega \subseteq \mathbb{Z}^n$, and let

$$T : \mathcal{D}(T) \subseteq \mathcal{L}^2(\Omega) \rightarrow \mathcal{L}^2(\Omega)$$

be a linear operator with domain $\mathcal{D}(T)$. We say that T is **self-adjoint** if it satisfies:

$$T = T^* \quad \text{and} \quad \mathcal{D}(T) = \mathcal{D}(T^*),$$

where T^* denotes the adjoint operator of T , defined by

$$\langle Tu, v \rangle = \langle u, T^* v \rangle \quad \text{for all } u \in \mathcal{D}(T), v \in \mathcal{D}(T^*).$$

Example 4 (Discrete Laplacian is self-adjoint). Let $\Omega \subseteq \mathbb{Z}$ be shift-invariant under ± 1 (e.g. $\Omega = \mathbb{Z}$ or periodic boundary conditions on a finite lattice), and consider the Hilbert space $\mathcal{L}^2(\Omega)$ with inner product

$$\langle f, g \rangle := \sum_{x \in \Omega} f(x) \overline{g(x)}.$$

Define the 1D discrete Laplacian by

$$\nabla^2 u := Eu + E^{-1}u - 2u, \quad \text{i.e.} \quad \nabla^2 = E + E^{-1} - 2I,$$

where $E^k u = u(x + k)$.

Since $E^* = E^{-1}$ (and hence $(E^{-1})^* = E$), we have

$$(\nabla^2)^* = (E + E^{-1} - 2I)^* = E^* + (E^{-1})^* - 2I = E^{-1} + E - 2I = \nabla^2.$$

Equivalently, for all $u, v \in \mathcal{L}^2(\Omega)$,

$$\begin{aligned} \langle \nabla^2 u, v \rangle &= \langle Eu, v \rangle + \langle E^{-1}u, v \rangle - 2\langle u, v \rangle \\ &= \langle u, E^{-1}v \rangle + \langle u, Ev \rangle - 2\langle u, v \rangle = \langle u, (E^{-1} + E - 2I)v \rangle = \langle u, \nabla^2 v \rangle. \end{aligned}$$

Hence ∇^2 is self-adjoint on $\mathcal{L}^2(\Omega)$.

Example 5 (Discrete n -Dimensional Laplacian is self-adjoint). Let $\Omega \subseteq \mathbb{Z}^n$ be shift-invariant under $\pm \mathbf{e}_i$ for $i = 1, \dots, n$ (e.g. $\Omega = \mathbb{Z}^n$ or periodic boundary conditions on a finite lattice), and consider the Hilbert space $\mathcal{L}^2(\Omega)$ with inner product

$$\langle f, g \rangle := \sum_{\mathbf{x} \in \Omega} f(\mathbf{x}) \overline{g(\mathbf{x})}.$$

For each coordinate i , let E_{x_i} denote the unit shift in the i -th direction:

$$E_{x_i} u := u(\mathbf{x} + \mathbf{e}_i), \quad E_{x_i}^{-1} u := u(\mathbf{x} - \mathbf{e}_i).$$

Define the discrete Laplacian by

$$\nabla^2 u := \sum_{i=1}^n [E_{x_i} u + E_{x_i}^{-1} u - 2u] \iff \nabla^2 = \sum_{i=1}^n (E_{x_i} + E_{x_i}^{-1} - 2I).$$

Using $(E_{x_i})^* = E_{x_i}^{-1}$ (hence $(E_{x_i}^{-1})^* = E_{x_i}$), we have

$$(\nabla^2)^* = \sum_{i=1}^n ((E_{x_i})^* + (E_{x_i}^{-1})^* - 2I) = \sum_{i=1}^n (E_{x_i}^{-1} + E_{x_i} - 2I) = \nabla^2.$$

Equivalently, for all $u, v \in \mathcal{L}^2(\Omega)$,

$$\langle \nabla^2 u, v \rangle = \sum_{i=1}^n (\langle E_{x_i} u, v \rangle + \langle E_{x_i}^{-1} u, v \rangle - 2\langle u, v \rangle) = \sum_{i=1}^n (\langle u, E_{x_i}^{-1} v \rangle + \langle u, E_{x_i} v \rangle - 2\langle u, v \rangle) = \langle u, \nabla^2 v \rangle.$$

Hence ∇^2 is self-adjoint on $\mathcal{L}^2(\Omega)$.

3.4. The Kronecker Delta Function

Definition 49 (Kronecker Delta Function). The Kronecker delta function is defined as

$$\delta(x - a) = \begin{cases} 1, & \text{if } x = a \\ 0, & \text{otherwise} \end{cases}$$

for any $x, a \in \mathbb{Z}$.

Definition 50 (Multivariable Kronecker Delta Function). Let $\delta : \mathbb{Z}^n \rightarrow \mathbb{R}$ be defined, for $\mathbf{x} = (x_1, \dots, x_n)$ and $\mathbf{s} = (s_1, \dots, s_n) \in \mathbb{Z}^n$, by

$$\delta(x_1 - s_1, \dots, x_n - s_n) := \prod_{i=1}^n \delta(x_i - s_i),$$

where $\delta(x)$ is the one-dimensional Kronecker delta function

Proposition 3 (Delta Representation of Discrete Functions). Let $\Omega = \mathbb{Z}^n$ and consider the Hilbert space $\mathcal{L}^2(\Omega)$ with inner product

$$\langle f, g \rangle := \sum_{\mathbf{x} \in \Omega} f(\mathbf{x}) \overline{g(\mathbf{x})}.$$

For each $\mathbf{s} \in \mathbb{Z}^n$, define the multivariable Kronecker delta

$$\delta(\mathbf{x} - \mathbf{s}) := \prod_{i=1}^n \delta(x_i - s_i).$$

Then every $f \in \mathcal{L}^2(\mathbb{Z}^n)$ can be represented as

$$f(\mathbf{x}) = \sum_{\mathbf{s} \in \mathbb{Z}^n} f(\mathbf{s}) \delta(\mathbf{x} - \mathbf{s}),$$

where the series converges in \mathcal{L}^2 . Moreover, the family $\{\delta(\mathbf{x} - \mathbf{s}) : \mathbf{s} \in \mathbb{Z}^n\}$ forms an orthonormal basis of $\mathcal{L}^2(\mathbb{Z}^n)$.

Proof. For $\mathbf{m}, \mathbf{n} \in \mathbb{Z}^n$,

$$\langle \delta(\mathbf{x} - \mathbf{m}), \delta(\mathbf{x} - \mathbf{n}) \rangle = \sum_{\mathbf{x} \in \mathbb{Z}^n} \delta(\mathbf{x} - \mathbf{m}) \overline{\delta(\mathbf{x} - \mathbf{n})}.$$

The product is nonzero only when $\mathbf{x} = \mathbf{m} = \mathbf{n}$, hence

$$\langle \delta(\mathbf{x} - \mathbf{m}), \delta(\mathbf{x} - \mathbf{n}) \rangle = \begin{cases} 1, & \mathbf{m} = \mathbf{n}, \\ 0, & \mathbf{m} \neq \mathbf{n}. \end{cases}$$

Thus they are mutually orthogonal and each satisfies

$$\|\delta(\mathbf{x} - \mathbf{s})\|_2^2 = \langle \delta(\mathbf{x} - \mathbf{s}), \delta(\mathbf{x} - \mathbf{s}) \rangle = 1,$$

so the set is orthonormal. Finally, for any $f \in \mathcal{L}^2(\mathbb{Z}^n)$,

$$f(\mathbf{x}) = \sum_{\mathbf{s} \in \mathbb{Z}^n} \langle f, \delta(\mathbf{x} - \mathbf{s}) \rangle \delta(\mathbf{x} - \mathbf{s}),$$

and $\langle f, \delta(\mathbf{x} - \mathbf{s}) \rangle = f(\mathbf{s})$, which completes the proof. \square

Example 6 (One-dimensional case). When $n = 1$, the multivariable Kronecker delta reduces to the standard one-dimensional Kronecker delta

$$\delta(x - s) := \begin{cases} 1, & x = s, \\ 0, & x \neq s, \end{cases} \quad x, s \in \mathbb{Z}.$$

In $\mathcal{L}^2(\mathbb{Z})$ with inner product

$$\langle f, g \rangle := \sum_{x \in \mathbb{Z}} f(x) \overline{g(x)},$$

the family $\{\delta(x - s) : s \in \mathbb{Z}\}$ satisfies

$$\langle \delta(x - m), \delta(x - n) \rangle = \begin{cases} 1, & m = n, \\ 0, & m \neq n, \end{cases}$$

so it forms an orthonormal basis of $\mathcal{L}^2(\mathbb{Z})$. Every $f \in \mathcal{L}^2(\mathbb{Z})$ admits the representation

$$f(x) = \sum_{s \in \mathbb{Z}} f(s) \delta(x - s),$$

where the series converges in \mathcal{L}^2 .

3.5. The Discrete Green's Function

Definition 51 (Discrete Green's Function). Let $\Omega \subseteq \mathbb{Z}^n$, and let

$$\mathbf{x} = (x_1, x_2, \dots, x_n) \in \Omega.$$

Consider a linear partial difference equation

$$L(u) = f(t, \mathbf{x}),$$

where t is the time variable, L is a linear shift operator acting on the spatial variables, and the equation is subject to certain boundary conditions on Ω .

The discrete Green's function $G(t, \mathbf{x})$ is the solution of the equation

$$L(G) = 0, \quad t > 0,$$

with the impulse initial condition

$$G(0, \mathbf{x}) = \delta(\mathbf{x}),$$

where $\delta(\mathbf{x})$ is the multivariable Kronecker delta centered at the origin.

Proposition 4 (Solution as Convolution of the Green's Function). Let $\Omega \subseteq \mathbb{Z}^n$ and consider the linear partial difference equation

$$L(u) = f(t, \mathbf{x}), \quad \mathbf{x} \in \Omega, \quad t \geq 0,$$

where L is a linear shift operator in the spatial variables. Let $G(t, \mathbf{x})$ be the discrete Green's function, i.e., the solution of $L(G) = 0$ with $G(0, \mathbf{x}) = \delta(\mathbf{x})$.

Then the unique solution $u(t, \mathbf{x})$ subject to zero initial condition is given by

$$u(t, \mathbf{x}) = \sum_{\tau=0}^t \sum_{\mathbf{s} \in \Omega} G(t - \tau, \mathbf{x} - \mathbf{s}) f(\tau, \mathbf{s}).$$

Proof of "Solution as Convolution of the Green's Function". Fix $\Omega \subseteq \mathbb{Z}^n$. Let L be a linear, causal evolution operator that is shift-invariant in space and time, and let $G(t, \mathbf{x})$ denote the discrete Green's function, i.e. the response to a unit space-time impulse at the origin:

$$\text{input } \delta(t) \delta(\mathbf{x}) \mapsto \text{output } G(t, \mathbf{x}), \quad t \geq 0.$$

Step 1 (delta expansion of the forcing). For any forcing $f : \mathbb{Z}_{\geq 0} \times \Omega \rightarrow \mathbb{C}$ we have the (space-time) delta representation

$$f(t, \mathbf{x}) = \sum_{\tau=0}^{\infty} \sum_{\mathbf{s} \in \Omega} f(\tau, \mathbf{s}) \delta(t - \tau) \delta(\mathbf{x} - \mathbf{s}),$$

with convergence in $\mathcal{L}^2(\Omega)$ for each fixed t (or in a suitable function space depending on the problem).

Step 2 (responses to elementary impulses). Let $e_{\tau, \mathbf{s}}(t, \mathbf{x}) := \delta(t - \tau) \delta(\mathbf{x} - \mathbf{s})$. By time- and space-shift invariance of L , the response to $e_{\tau, \mathbf{s}}$ is the shifted Green's function

$$u_{\tau, \mathbf{s}}(t, \mathbf{x}) = G(t - \tau, \mathbf{x} - \mathbf{s}) \quad (\text{understood as 0 for } t < \tau).$$

Step 3 (linear superposition). By linearity of L , the solution to $L(u) = f$ with zero initial condition is the superposition of the elementary responses weighted by the coefficients $f(\tau, \mathbf{s})$ from Step 1:

$$u(t, \mathbf{x}) = \sum_{\tau=0}^t \sum_{\mathbf{s} \in \Omega} f(\tau, \mathbf{s}) G(t - \tau, \mathbf{x} - \mathbf{s}).$$

(The upper limit t reflects causality; terms with $\tau > t$ vanish.)

Thus $u = G * f$ is the discrete space-time convolution of the Green's function with the forcing, which proves the stated formula. \square

3.6. Boundary Value Problems

We restrict to the one-dimensional case with spatial index $x \in \{L, L+1, \dots, M\}$ and unit grid spacing. In the discrete setting, the following are direct analogues of the classical boundary conditions for continuous PDE, obtained by replacing derivatives with finite difference operators.

Dirichlet boundary condition.

Given constants $a, b \in \mathbb{C}$, the Dirichlet condition fixes the value of u at both ends:

$$u(t, L) = a, \quad u(t, M) = b.$$

Physically, this corresponds to prescribing a fixed state (e.g. temperature, displacement) at the boundary points.

Neumann boundary condition.

In the continuous PDE setting, a Neumann boundary condition prescribes the normal derivative of the solution at the boundary:

$$\partial_x u|_{x=L} = a, \quad \partial_x u|_{x=M} = b,$$

where $a, b \in \mathbb{C}$ are prescribed constants. Physically, the Neumann condition represents a prescribed flux across the boundary. For example, in the heat equation, $\partial_x u$ corresponds to the heat flux according to Fourier's law.

In the context of partial difference equations, we replace the spatial derivative with the discrete central difference operator Δ_x^c (unit grid spacing) and introduce ghost points:

$$\begin{aligned} \Delta_x^c u|_{x=L} &= \frac{u(t, L+1) - u(t, L-1)}{2} = a, \\ \Delta_x^c u|_{x=M} &= \frac{u(t, M+1) - u(t, M-1)}{2} = b, \end{aligned}$$

where $u(t, L-1)$ and $u(t, M+1)$ lie outside the computational domain. Solving these equations for the ghost points yields:

$$u(t, L-1) = u(t, L+1) - 2a, \quad u(t, M+1) = u(t, M-1) + 2b.$$

This “ghost point” technique allows the central difference stencil to be applied at the boundary while preserving second-order accuracy.

Robin boundary condition.

In the continuous PDE setting, a Robin boundary condition is a linear combination of the normal derivative and the function value:

$$\frac{\partial u}{\partial x} + p(x) u = q(x), \quad x \in \{L, M\},$$

where $p(x)$ and $q(x)$ are prescribed functions on the boundary (possibly constants). Physically, Robin conditions often model convective exchange or spring-type constraints.

In the discrete setting, we replace ∂_x with the central difference operator Δ_x^c :

$$\Delta_x^c u + p(x) u = q(x), \quad x \in \{L, M\}.$$

Using ghost points, for example at $x = L$:

$$\frac{u(t, L+1) - u(t, L-1)}{2} + p(L) u(t, L) = q(L),$$

so the ghost point $u(t, L-1)$ can be eliminated as

$$u(t, L-1) = u(t, L+1) - 2(q(L) - p(L) u(t, L)).$$

A similar formula holds at $x = M$. This preserves the central difference stencil at the boundary while enforcing the Robin condition.

4. Elementary Cellular Automata

4.1. Introduction

The study of one-dimensional cellular automata (1D CA) was significantly advanced by Stephen Wolfram [7], who systematically explored the behaviour of all 256 elementary rules. These models, despite their simplicity, can generate a wide range of complex spatiotemporal patterns, including periodic structures, nested fractals, localized structures, and even chaotic behaviour.

All 1D cellular automata can be expressed as **autonomous partial difference equations**, meaning that the update rule does not include any external forcing term. Moreover, most of these equations are inherently nonlinear, due to the logical (Boolean) nature of the local interactions.

In this section, we formulate several 1D cellular automata as partial difference equations. Some of these formulations are based on known Boolean algebra transformations, converting logical rules into Boolean polynomials, and then into partial difference equations. Others are derived heuristically by observing the pattern of evolution.

This reformulation has several important advantages:

- It provides a unified mathematical framework to study discrete systems.
- It allows the application of analytical tools such as operator theory, stability analysis, and even Green's functions.
- It enables the exploration of different types of initial and boundary conditions in a more formalized way.
- It makes it possible to introduce non-autonomous terms and study how external forcing influences the evolution.

This approach transforms cellular automata from mere computational toys into objects of rigorous mathematical investigation within the broader context of discrete dynamical systems.

4.2. Notation Evolution

In the work by Itoh and Chua [8], one-dimensional cellular automata were expressed in the form of difference equations using the notation x_i^n . For example, Rule 90 can be written as:

$$x_i^{n+1} = x_{i-1}^n + x_{i+1}^n \pmod{2}.$$

However, they did not explicitly introduce the concept of *partial difference equations*, and continued to refer to such formulations simply as “difference equations.”

In contrast, in the present work we explicitly adopt the notion of *partial difference equations*. My earlier notation was, for example, in the case of Rule 90:

$$u(t+1, x) = u(t, x-1) + u(t, x+1) \pmod{2}.$$

Although clear for low-dimensional cases, this style becomes cumbersome when multiple spatial variables are involved. For example, with $u(t, x, y, z)$, the equations become long, cluttered, and visually unappealing. Moreover, in more complex models such as the Game of Life or the forest fire

model, I initially used custom functions, resulting in update rules that were messy and syntactically similar to programming code rather than mathematical expressions.

To address this, I spent several months carefully defining a set of reusable building blocks, including the Kronecker delta function, Heaviside step function, modulo function, shift operators, difference operators, and the discrete Laplacian. With these tools, the notation becomes significantly more compact and expressive.

For example, Rule 90 can now be written in operator form as:

$$E_t u = \text{mod}_2(E_x u + E_x^{-1} u), \quad u : \mathbb{Z}^2 \rightarrow \mathbb{Z}_2.$$

The advantages are even more apparent in higher dimensions. Consider the two-dimensional discrete diffusion equation in traditional notation:

$$u(t+1, x, y) = u(t, x, y) + (u(t, x-1, y) + u(t, x+1, y) + u(t, x, y-1) + u(t, x, y+1) - 4u(t, x, y)),$$

which is long and cumbersome. In operator notation, this simply becomes:

$$E_t u = u + \nabla^2 u,$$

or equivalently,

$$\Delta_t u = \nabla^2 u.$$

This operator-based notation is not only more concise, but also opens the door to applying discrete functional analysis, operator theory, and other mathematical tools to the study of partial difference equations.

4.3. The Rule 90

The **Rule 90** cellular automaton can be written as a nonlinear partial difference equation of the form:

$$E_t u = \text{mod}_2(E_x^{-1} u + E_x u)$$

where $\text{mod}_2(x)$ is defined as:

$$\text{mod}_2(x) := \begin{cases} 1, & \text{if } x \text{ is odd} \\ 0, & \text{if } x \text{ is even} \end{cases}$$

This function is clearly non-linear, making the equation itself non-linear.

We refer to this equation as the **Sierpiński Equation**, due to its deep connection with the Sierpiński triangle.

Define a discrete delta function as:

$$\delta(x-a) := \begin{cases} 1, & \text{if } x = a \\ 0, & \text{otherwise} \end{cases}$$

Given the initial condition $u(0, x) = \delta(x)$ and no boundary condition, the exact solution to the equation is:

$$u(t, x) = \text{mod}_2(C(2t, x+t))$$

where $C(x, y)$ is the binomial coefficient, defined as:

$$C(x, y) := \begin{cases} \frac{x!}{(x-y)!y!}, & \text{if } 0 \leq y \leq x \\ 0, & \text{otherwise} \end{cases}$$

For a general initial condition $u(0, x) = f(x)$, the solution becomes:

$$u(t, x) = \text{mod}_2 \left(\sum_{s \in \mathbb{Z}} f(s) \cdot C(2t, x + t - s) \right)$$

This system exhibits **chaotic behaviour** in a discrete binary field. Remarkably, despite its chaotic dynamics, we have found a closed-form analytical solution. This provides a foundation to define and analyze concepts like **chaotic functions** and **quasichaotic functions** within the framework of discrete functional analysis.

Example 1

This is the spatiotemporal plot of Rule 90 with the initial condition $u(0, x) = \delta(x)$ and **no boundary conditions**.

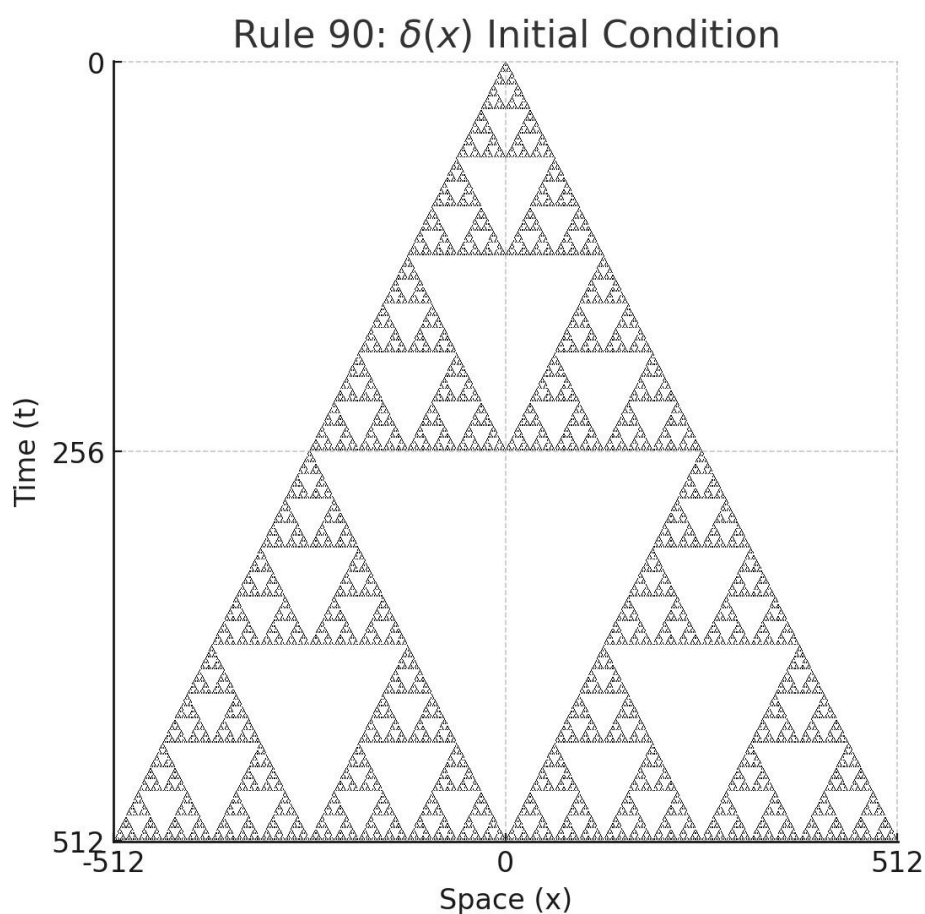


Figure 2. Spatiotemporal plot of Rule 90 with $u(0, x) = \delta(x)$ and no boundary conditions.

It clearly forms a Sierpiński triangle. Note that each cell's state depends only on its left and right neighbours. Thus, this fractal is not globally planned, but emerges from local interactions.

Example 2

This is the spatiotemporal plot of Rule 90 with the initial condition $u(0, x) = \delta(x)$ and boundary conditions $u(t, -100) = u(t, 100) = 0$.

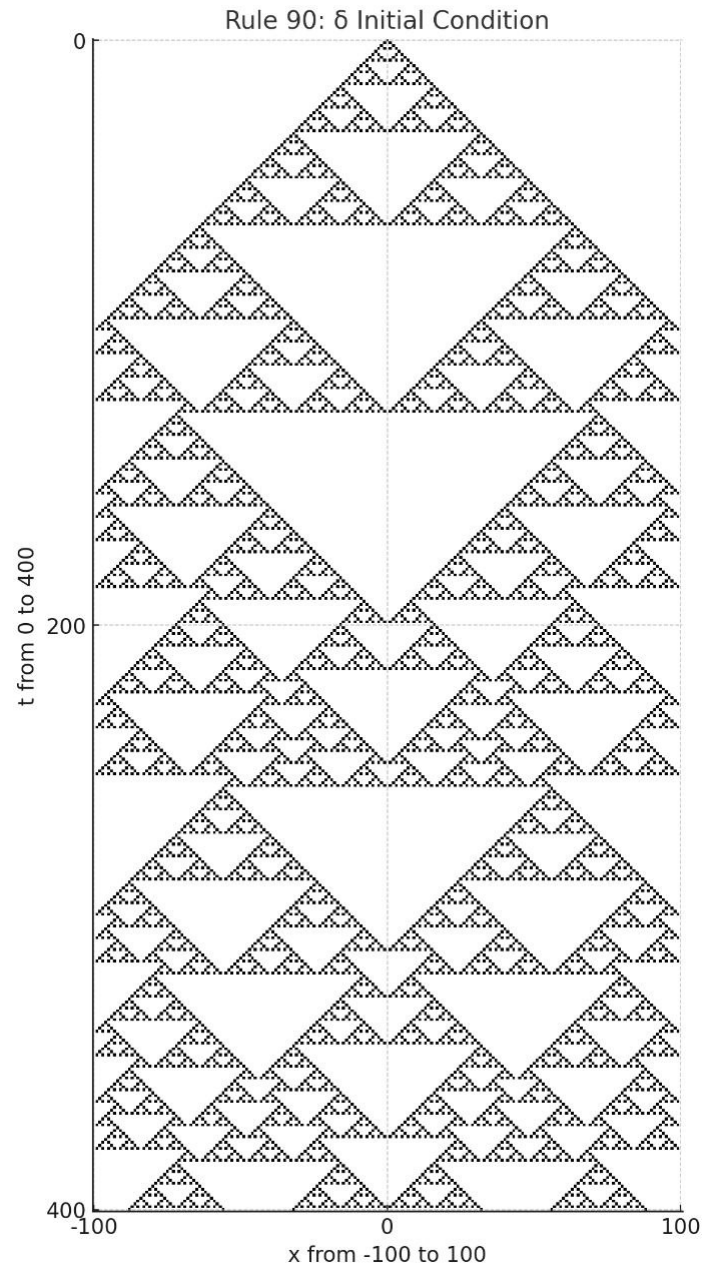


Figure 3. Spatiotemporal plot of Rule 90 with $u(0, x) = \delta(x)$ and boundary conditions $u(t, -100) = u(t, 100) = 0$.

Initially, we observe a perfect Sierpiński triangle. However, once the pattern reaches the boundary, it breaks down into spatiotemporal chaos and becomes unpredictable.

Example 3

This is the spatiotemporal plot of Rule 90 with **random initial condition** and boundary conditions $u(t, -200) = u(t, 200) = 0$.

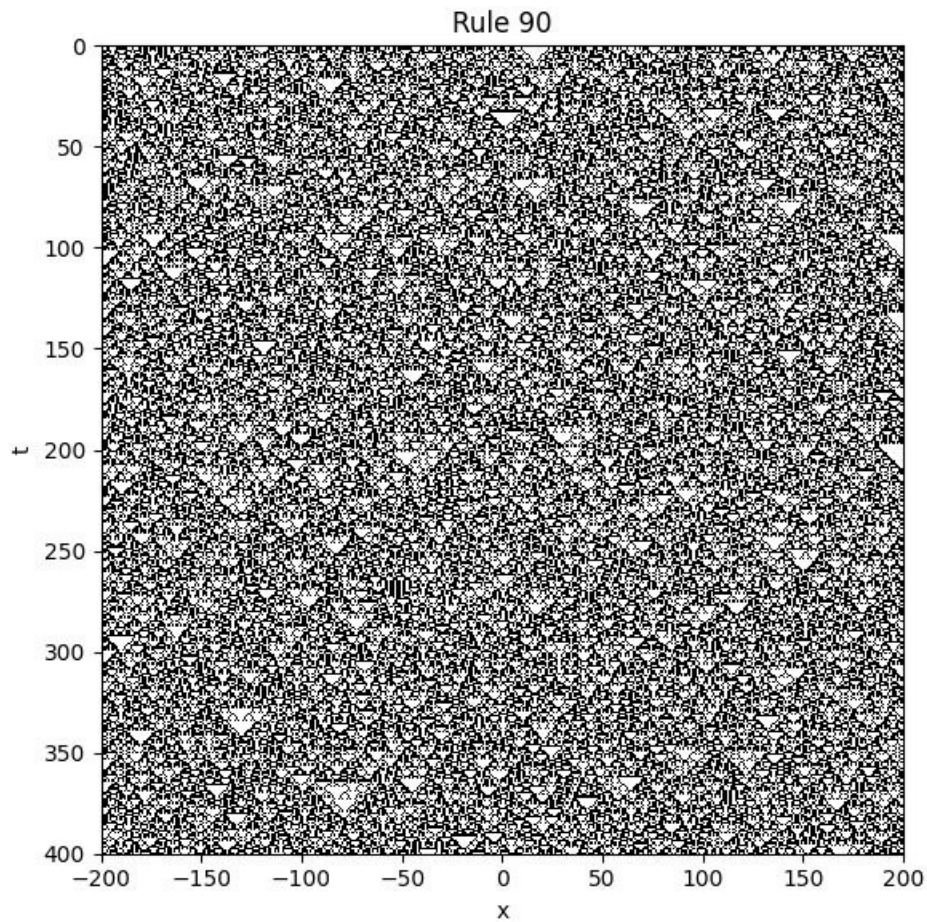


Figure 4. Spatiotemporal plot of Rule 90 with random initial condition and boundary conditions.

As shown, the system exhibits spatiotemporal chaos.

4.4. Rule 30

The evolution equation for Rule 30, derived using Boolean algebra, is given by:

$$E_t u = \text{mod}_2 \left(E_x^{-1} u + E_x u + u + u \cdot E_x u \right),$$

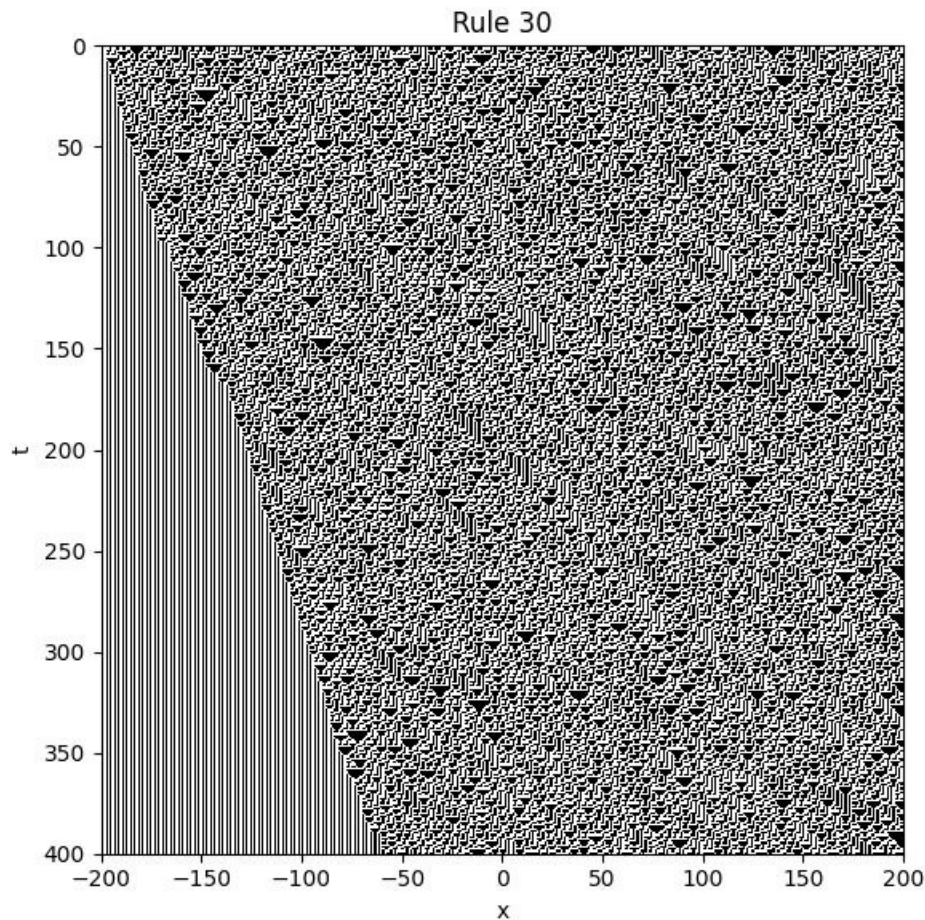
where E_t is the time shift operator, E_x is the spatial right shift operator.

Example

We consider the following simulation setup:

- **Initial condition:** $u(0, x)$ is chosen randomly with values in $\{0, 1\}$.
- **Boundary condition:** $u(t, -200) = u(t, 200) = 0$ for all t .

The figure below shows the spatiotemporal evolution of the system under Rule 30:



It is evident that the system exhibits *spatiotemporal chaos*, characterized by aperiodic and unpredictable patterns across both space and time.

4.5. The Rule 153

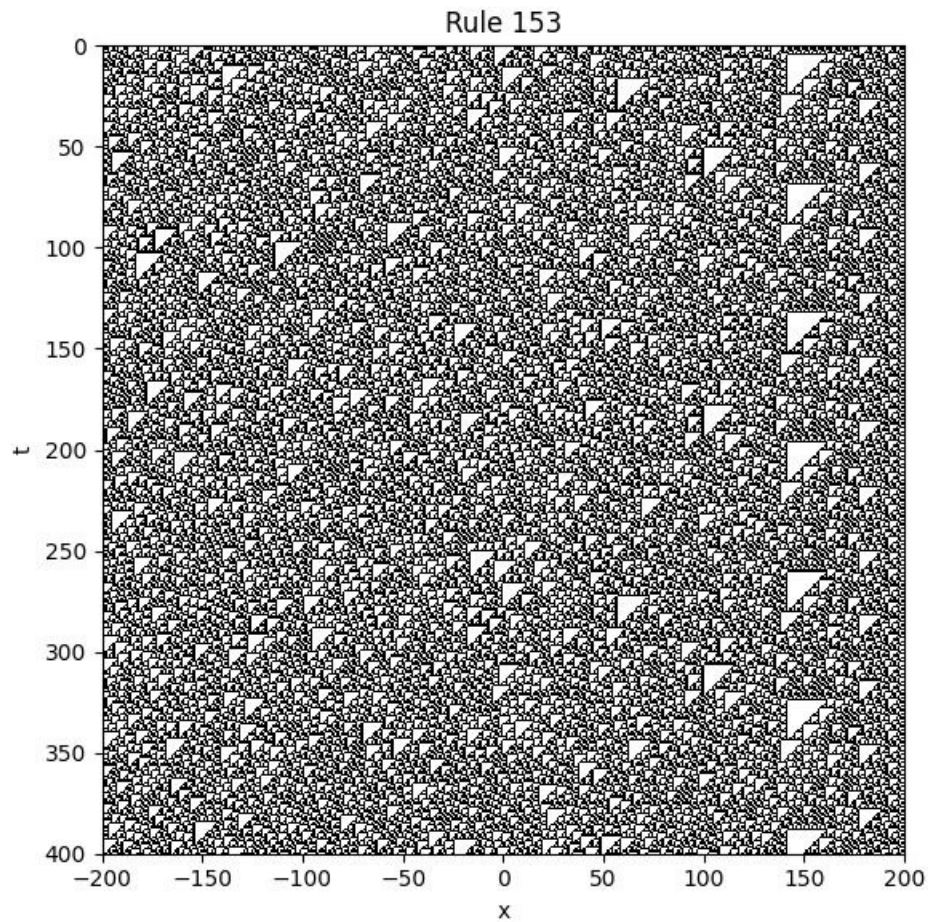
We define the evolution equation as follows:

$$E_t u = \text{mod}_2 \left(\left\lfloor \frac{1 - E_x^{-1} u}{1 + E_x u} + E_x u \cdot E_x^{-1} u + \left| 3E_x^{-1} u - 4E_x u \right| + u \right\rfloor \right)$$

Example

- **Initial condition:** $u(0, x)$ is assigned a random binary value (0 or 1) at each spatial point.
- **Boundary condition:** $u(t, -200) = u(t, 200) = 0$ for all t .
- **Colour scheme:** 0 is shown as white, and 1 is shown as black.

The following image illustrates the space-time evolution from $t = 0$ to $t = 400$, and $x \in [-200, 200]$, with a random initial condition:



This system exhibits spatiotemporal chaos.

4.6. Rule 45

The Boolean update rule for Rule 45 can be written as the following equation:

$$E_t u = \text{mod}_2 \left(1 + u + E_x^{-1} u + u \cdot E_x u \right)$$

Example

Initial condition: $u(0, x)$ is random.

Boundary conditions: $u(t, -200) = u(t, 200) = 0$.

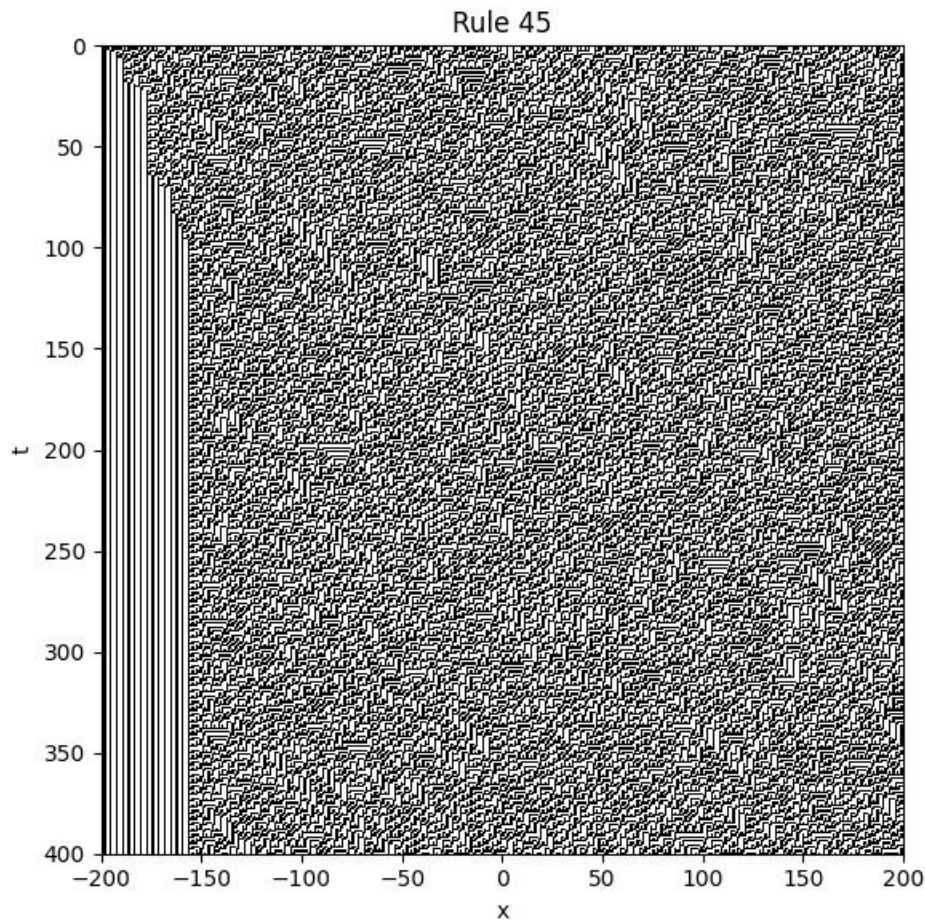


Figure 5. Spatiotemporal plot of Rule 45 with random initial condition and fixed boundary conditions.

As shown in the figure, this system also exhibits spatiotemporal chaos. The resulting pattern features strange, curled, filament-like structures.

4.7. Rule 105

The Boolean update rule for Rule 105 can be written using logical operations as follows:

$$E_t u = \text{mod}_2 \left(E_x^{-1} u \cdot u \cdot \delta(E_x u) + E_x^{-1} u \cdot \delta(u) \cdot E_x u + \delta(E_x^{-1} u) \cdot u \cdot E_x u + \delta(E_x^{-1} u) \cdot \delta(u) \cdot \delta(E_x u) \right)$$

Here, $\delta(x)$ denotes the Boolean NOT operation.

Example

Initial condition: $u(0, x)$ is random.

Boundary conditions: $u(t, -200) = u(t, 200) = 0$.

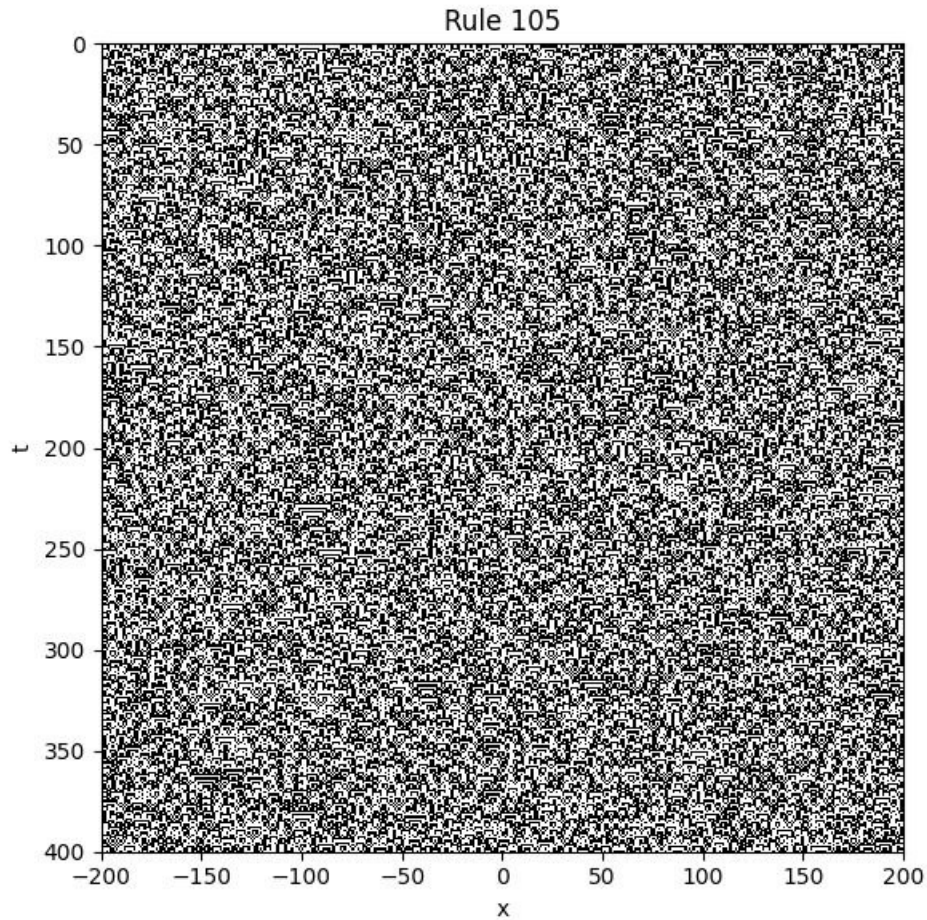


Figure 6. Spatiotemporal plot of Rule 105 with random initial condition and fixed boundaries.

As shown in the figure, this system also exhibits spatiotemporal chaos. The resulting pattern consists of intricate structures resembling scales or rippling textures.

4.8. Rule 169

The Boolean update rule for Rule 169 can be expressed as:

$$E_t u = \text{mod}_2 \left(E_x^{-1} u \cdot u + E_x^{-1} u + u + E_x u + 1 \right)$$

Example

Initial condition: $u(0, x)$ is random.

Boundary conditions: $u(t, -200) = u(t, 200) = 0$.

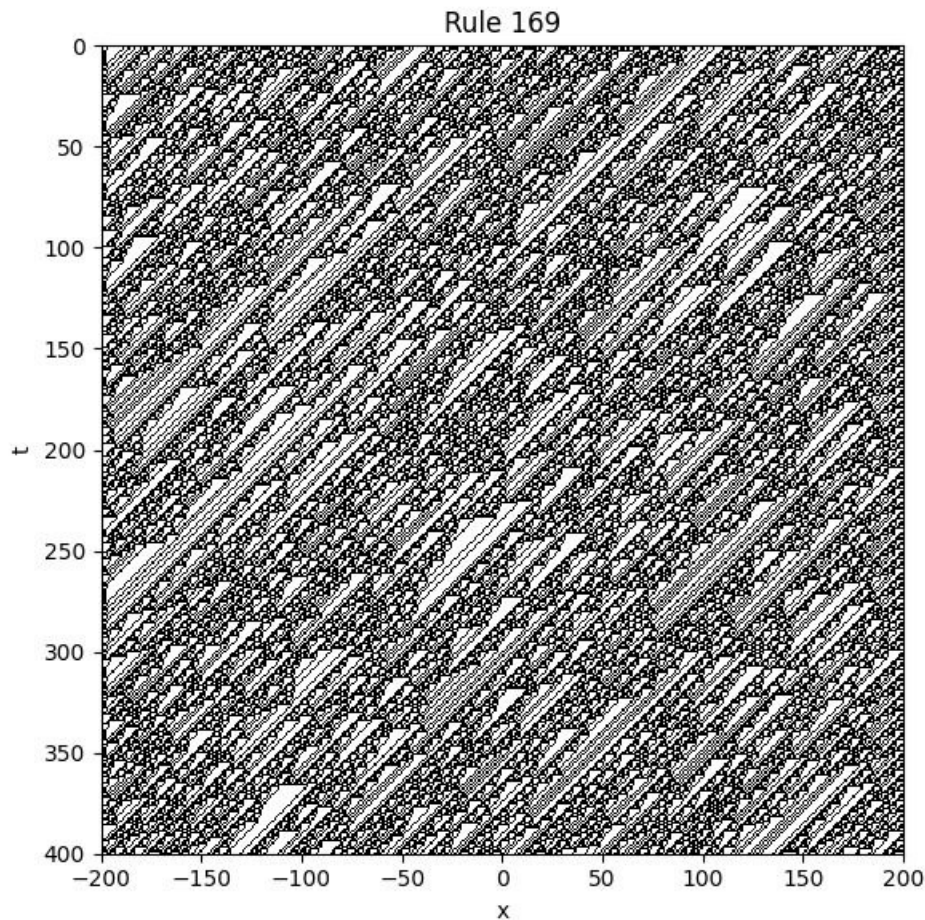


Figure 7. Spatiotemporal plot of Rule 169 with random initial condition and fixed boundaries.

As shown in the figure, the system exhibits spatiotemporal chaos. The resulting patterns resemble diagonal stalactites, indicating complex local interactions and emergent behaviour.

4.9. Rule 210

The Boolean update rule for Rule 210 is given by:

$$E_t u = \text{mod}_2 \left(E_x^{-1} u \cdot u \cdot E_x u + E_x^{-1} u \cdot u \cdot \delta(E_x u) + E_x^{-1} u \cdot \delta(u) \cdot \delta(E_x u) + \delta(E_x^{-1} u) \cdot \delta(u) \cdot E_x u \right)$$

Example

Initial condition: $u(0, x)$ is random.

Boundary conditions: $u(t, -200) = u(t, 200) = 0$.

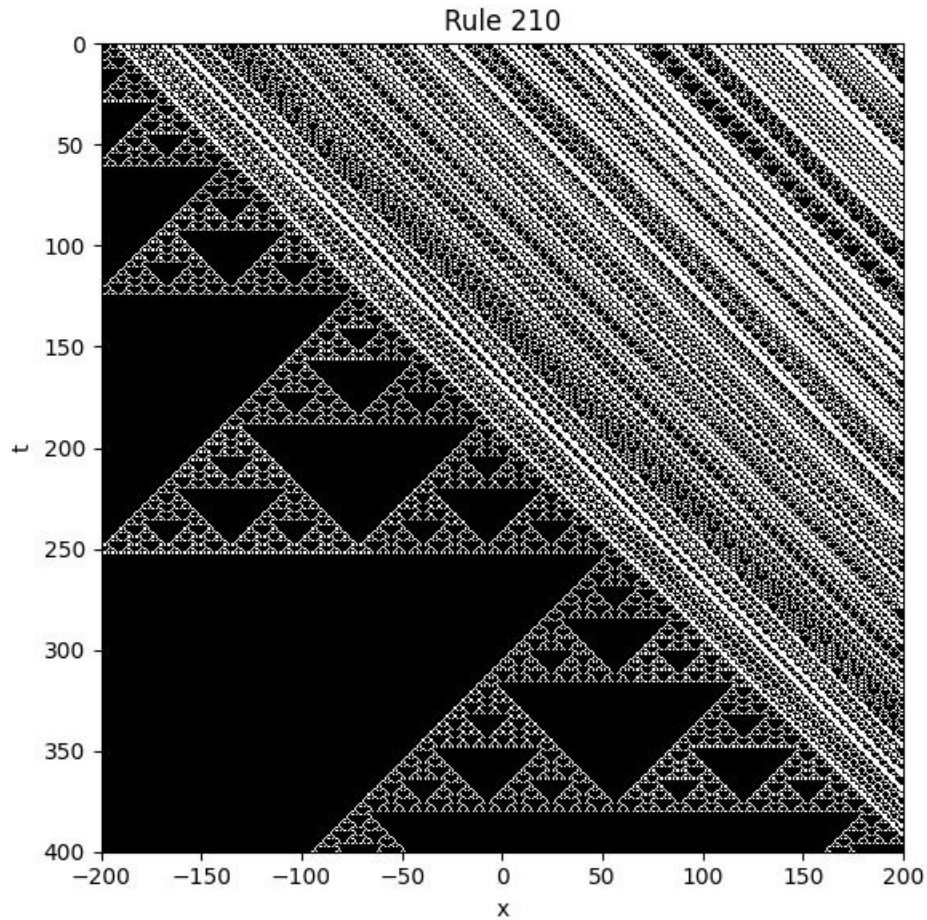


Figure 8. Spatiotemporal plot of Rule 210 with random initial condition and fixed boundaries.

As shown in the figure, the system exhibits a highly unusual structure: one region resembles a Sierpiński triangle, while the other shows diagonally arranged motifs. This dual pattern reflects complex local interactions and potentially bifurcating spatial dynamics.

4.10. Rule 137

We define the evolution equation as follows:

$$E_t u = \text{mod}_2 \left(\left| \left(E_x^{-1} u - \pi^u \right) (\pi + u + E_x u) \right| + \left| \frac{1-u}{1+u} \right| + \left| 3E_x^{-1} u - 4u \right| \right)$$

each state value satisfies

$$u(t, x) \in \{0, 1\}.$$

This rule is an example of a nonlinear partial difference equation, operating over a binary state space.

Interestingly, the system exhibits a mixture of order and apparent randomness. Some regions evolve into highly regular patterns, while others appear chaotic and unpredictable. This suggests that the system may operate near the *edge of chaos*, a regime known for its potential computational richness and complexity.

Although we do not currently possess a closed-form analytic solution to this equation, the intricate structure of its evolving patterns can still be studied using tools such as combinatorics, discrete functional analysis, and spatiotemporal statistics.

Example

Initial condition: $u(0, x)$ is random.

Boundary conditions: $u(t, -200) = u(t, 200) = 0$.

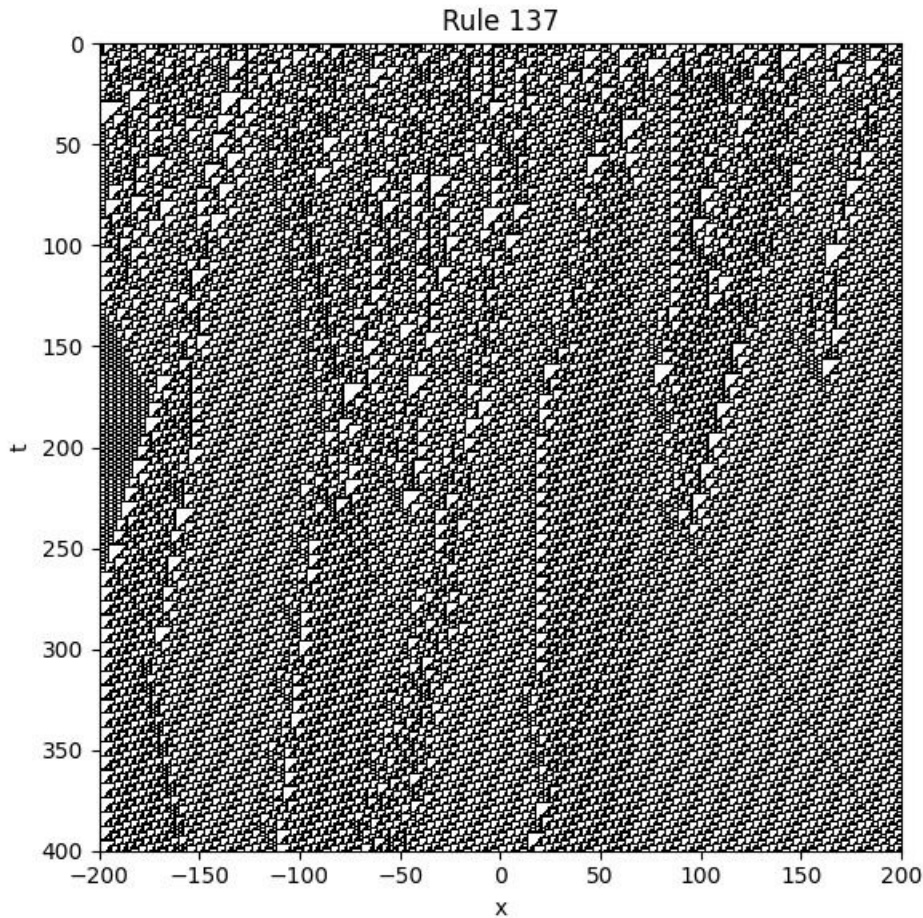


Figure 9. Spatiotemporal plot of Rule 137 with random initial condition and fixed boundaries.

The pattern generated by Rule 137 closely resembles that of Rule 110. It exhibits complex behaviour with both periodic islands and chaotic seas. There are large regions of crystalline structure, along with distinct triangular motifs, indicating rich dynamics and spatial self-organization.

4.11. Rule 110

Using Boolean algebra, we derive the evolution equation for Rule 110 as follows:

$$E_t u = \text{mod}_2 \left(u + E_x u + u \cdot E_x u + u \cdot E_x u \cdot E_x^{-1} u \right),$$

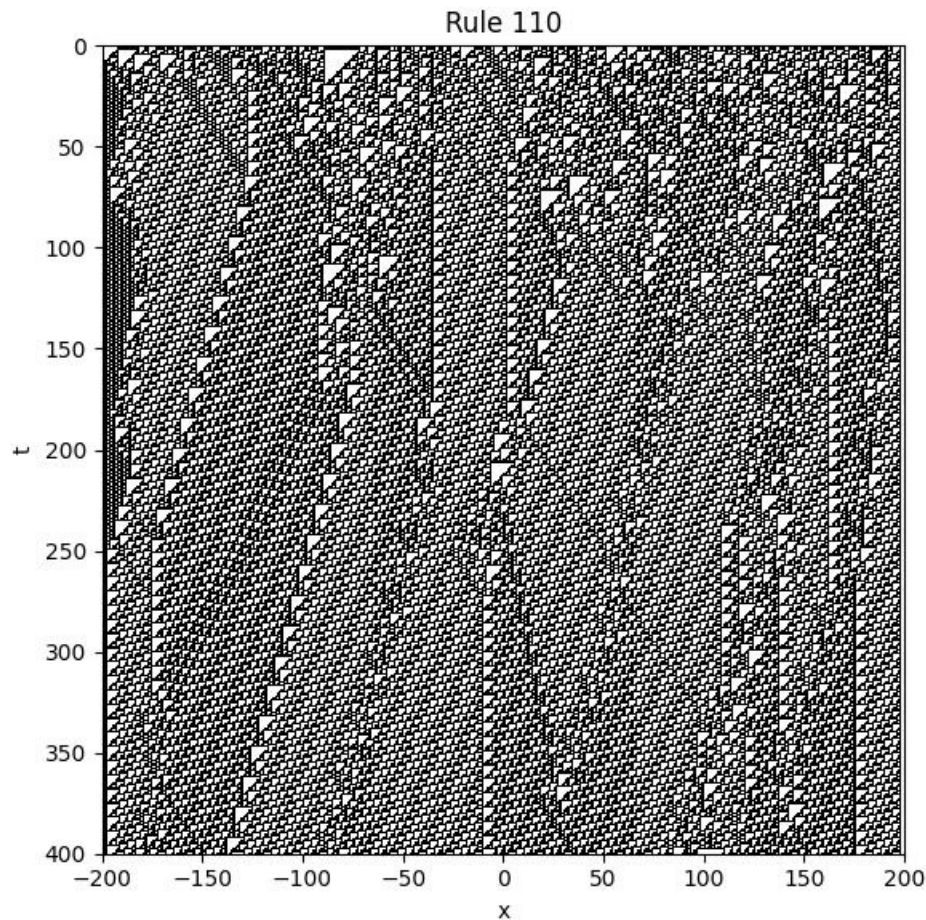
where E_t denotes the time shift operator, E_x is the spatial right-shift operator, and mod_2 denotes addition modulo 2.

Example

We consider the following setup:

- **Initial condition:** $u(0, x)$ is chosen randomly with values in $\{0, 1\}$.
- **Boundary condition:** $u(t, -200) = u(t, 200) = 0$ for all t .

The figure below illustrates the space-time evolution of the system under Rule 110:



5. Other 1D Nonlinear Partial Difference Equations

5.1. Introduction

Beyond classical two-state cellular automata such as Rule 90 or Rule 137, we have discovered that a wide variety of other one dimensional nonlinear partial difference equations (PΔE) can be constructed. These equations may involve arithmetic operations, logical operations, or floor and absolute value functions. They do not necessarily follow the strict rule-based framework of cellular automata, yet they remain fully deterministic and discrete in both space and time.

Such equations often exhibit rich and unexpected dynamical behaviour. Some generate spatially localized chaos, while others display periodic or quasiperiodic structures. This diversity suggests that one-dimensional nonlinear PΔE form a broader and more general class of systems than traditionally studied automata.

In the following subsections, we will explore several such examples and discuss their qualitative features.

5.2. Sierpinski Fan Equation

Consider the following nonlinear partial difference equation:

$$E_t u = \text{mod}_3 \left(E_x^{-1} u + u + E_x u \right)$$

with initial condition:

$$u(0, x) = \delta(x), \quad \text{and no boundary condition.}$$

$u = u(t, x) \in \{0, 1, 2\}$ Here, the delta function $\delta(x)$ is defined as:

$$\delta(x) = \begin{cases} 1, & x = 0 \\ 0, & x \neq 0 \end{cases}$$

This system produces a fascinating fractal structure resembling a "fan" or "tree", hence we refer to it as the Sierpinski Fan Equation.

Although we currently do not have an analytic solution to this equation, the emergent pattern clearly exhibits self-similarity and modular periodicity, which are typical features of modular arithmetic-driven cellular automata. This suggests a rich underlying structure that deserves further investigation.

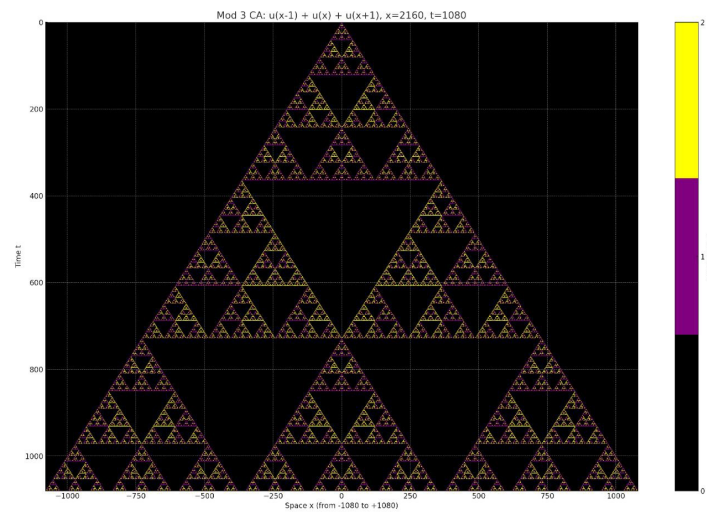


Figure 10. Spatiotemporal evolution of the Sierpinski Fan Equation

5.3. Mod 5 Sierpiński Equation

We define a new nonlinear partial difference equation as follows:

$$E_t u = \text{mod}_5 \left(E_x^{-2} u + E_x^{-1} u + u + E_x^1 u + E_x^2 u \right)$$

with the initial condition:

$$u(0, x) = \delta(x)$$

where:

- $\delta(x)$ is the discrete delta function defined by $\delta(x) = 1$ if $x = 0$, and 0 otherwise;
- $u = u(t, x) \in \{0, 1, 2, 3, 4\}$ for all t, x ;
- The system is updated on a discrete 1D lattice with no boundary condition.

This equation can be seen as a generalization of the Rule 90 or Sierpiński triangle in modulo 2. The modulo 5 version creates a far richer fractal structure.

From the spatiotemporal plot, we observe that instead of the classic single upside-down triangle in each upright triangle (as in the binary Sierpiński triangle), this pattern exhibits **ten** upside-down triangles within each upright triangle. This suggests a more intricate self-similar structure with higher-order modular symmetry.

We call this the **Mod 5 Sierpiński Equation**.

5.4. Mod 4 Sierpinski Fractal Equation

We consider the following nonlinear partial difference equation:

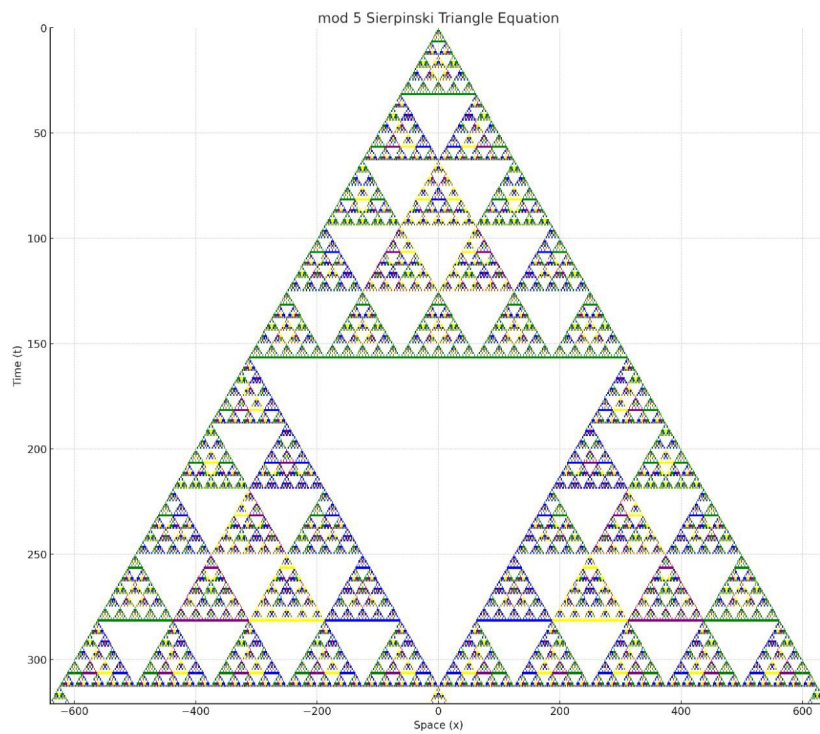


Figure 11. Spatiotemporal evolution of the Mod 5 Sierpiński Equation

$$E_t u = \text{mod}_4 \left(E_x^{-1} u + u + E_x u + \delta(x) \right)$$

with initial condition:

$$u(0, x) = \delta(x)$$

and no boundary condition.

$$u = u(t, x) \in \{0, 1, 2, 3\}$$

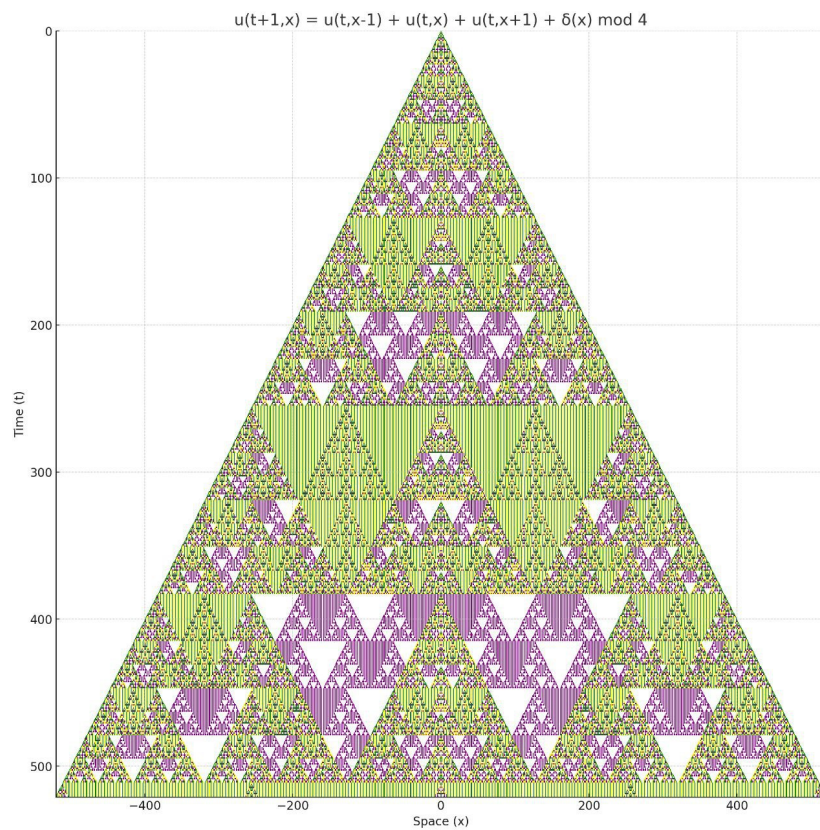
Here, $\delta(x)$ is the discrete delta function defined by:

$$\delta(x) = \begin{cases} 1, & \text{if } x = 0 \\ 0, & \text{otherwise} \end{cases}$$

The values of $u(t, x)$ are restricted to $\{0, 1, 2, 3\}$, and we visualize them using the following colour scheme:

Value	Colour
0	White
1	Green
2	Purple
3	Yellow

The spatiotemporal plot of this equation is shown below:



We observe that this system generates a strikingly complicated fractal structure, reminiscent of a Sierpinski triangle, but with richer periodic colour layers. Despite its simplicity, the system exhibits highly nontrivial dynamics that may reflect deep combinatorial or algebraic patterns.

6. Coupled Map Lattice

6.1. 1D Coupled Map Lattice

The one-dimensional Coupled Map Lattice (CML), originally proposed by Kaneko [9], is a prototypical model for spatiotemporal chaos. It is defined by the following recurrence relation:

$$u_s^{t+1} = (1 - \varepsilon)f(u_s^t) + \frac{\varepsilon}{2}(f(u_{s+1}^t) + f(u_{s-1}^t)),$$

where $u_s^t \in \mathbb{R}$, $f : \mathbb{R} \rightarrow \mathbb{R}$ is a nonlinear local map (e.g., logistic map), and $\varepsilon \in [0, 1]$ is the coupling strength.

We reformulate this equation using shift operators to obtain a compact operator-based representation:

$$E_t u = (1 - \varepsilon)f(u) + \frac{\varepsilon}{2}(f(E_x u) + f(E_x^{-1} u)),$$

where

$$u = u(t, x), \quad u : \mathbb{Z}^2 \rightarrow \mathbb{R}.$$

This equation is usually nonlinear autonomous Partial Difference Equation that captures spatiotemporal dynamics and is frequently used to model chaotic behaviour on discrete lattices.

6.2. 2D Coupled Map Lattice

The two-dimensional Coupled Map Lattice (2D CML) extends the 1D case to a square lattice, where each site interacts with its four nearest neighbours. The classical form is given by:

$$u_{i,j}^{t+1} = (1 - \varepsilon)f(u_{i,j}^t) + \frac{\varepsilon}{4} \left[f(u_{i+1,j}^t) + f(u_{i-1,j}^t) + f(u_{i,j+1}^t) + f(u_{i,j-1}^t) \right],$$

where $u_{i,j}^t \in \mathbb{R}$, $f: \mathbb{R} \rightarrow \mathbb{R}$, and $\varepsilon \in [0, 1]$ as before.

Using shift operators, the system can be rewritten in a compact operator-based form:

$$E_t u = (1 - \varepsilon)f(u) + \frac{\varepsilon}{4} \left[f(E_x u) + f(E_x^{-1} u) + f(E_y u) + f(E_y^{-1} u) \right],$$

where

$$u = u(t, x, y), \quad u: \mathbb{Z}^3 \rightarrow \mathbb{R}.$$

This operator representation highlights the spatial symmetry and facilitates generalization to higher dimensions, anisotropic coupling, or graph-based topologies.

6.3. Explanation

A *coupled map lattice* (CML) is a discrete-time, discrete-space dynamical system in which each lattice site evolves according to a prescribed local map and interacts with other sites via a coupling rule. The local map encodes the nonlinear dynamics at each site, while the coupling represents spatial interactions such as diffusion, transport, or synchronization.

CMLs are capable of producing *spatiotemporal chaos*, where complex, irregular patterns emerge and evolve across both space and time. They occupy an intermediate position between continuous partial differential equations and fully discrete cellular automata: like PDE, they describe the evolution of a field; like CA, they are inherently discrete in space and time.

In the context of this work, CMLs can be regarded as a particular subclass of partial difference equations in which the update operator contains a nonlinear local component together with a discrete coupling term. This perspective allows the mathematical machinery of discrete functional analysis to be applied directly to the study of CML dynamics.

7. The Conway's Game of Life

7.1. Conway's Equation

We express Conway's Game of Life [4] as a nonlinear autonomous partial difference equation:

$$E_t u = \delta(u - 1)[\delta(S_M u - 2) + \delta(S_M u - 3)] + \delta(u) \delta(S_M u - 3)$$

where the neighbourhood sum is defined as:

$$S_M u := \sum_{(i,j) \in M} E_x^i E_y^j u$$

with the following definitions:

- $u = u(t, x, y) \in \{0, 1\}$ is the binary state of the cell at time t and position (x, y) , where 1 denotes "alive" and 0 denotes "dead".
- E_t, E_x, E_y are the forward shift operators in time and spatial directions, respectively.
- M is the Moore neighbourhood excluding the center, i.e., the set of 8 nearest neighbours:

$$M = \{(-1, -1), (-1, 0), (-1, 1), (0, -1), (0, 1), (1, -1), (1, 0), (1, 1)\}$$

- $\delta(x)$ is the Kronecker delta function, defined as:

$$\delta(x) = \begin{cases} 1 & \text{if } x = 0 \\ 0 & \text{otherwise} \end{cases}$$

This formulation represents a **nonlinear autonomous partial difference equation**, where the next state of each cell depends only on its current state and the sum of its neighbours in the Moore neighbourhood.

7.2. Interpretation

We observe that: - **Still lifes** are the *steady-state solution*. - **Gliders** behave like *solitons*, stable moving structures. - **Oscillators** resemble *standing waves*, periodically repeating in place. - Each of these localized configurations can be seen as a **local attractor unit**. - Multiple local attractors can couple together, forming a **large-scale network** we call the *Attractor Coupling Network (ACN)*.

This structure is: - A *dynamic fractal network*, in the sense that it exhibits recursive nesting: a network within a network, evolving over time. - Capable of universal computation: Conway's Game of Life is known to be **Turing complete**.

This suggests a deep connection between simple discrete dynamics and the emergence of complex, structured, and computationally powerful behaviour.

8. Abelian Sandpile Model

8.1. Abelian Sandpile Equation

The evolution of the Abelian Sandpile Model [3] is given by the following equation:

$$E_t u = u - 4\theta(u - 4) + \sum_{(i,j) \in V} \theta(E_x^i E_y^j u - 4) + G(t, x, y)$$

where:

- $u(t, x, y) \in \{0, 1, 2, 3\}$
- $\theta(x)$ is the Heaviside step function:

$$\theta(x) = \begin{cases} 1, & \text{if } x \geq 0 \\ 0, & \text{otherwise} \end{cases}$$

- V is the von Neumann neighbourhood:

$$V = \{(1, 0), (-1, 0), (0, 1), (0, -1)\}$$

- $G(t, x, y)$ is the external forcing term (i.e., the addition of sand grains)

8.2. Remarks

This is a Non-Autonomous Partial Difference Equation.

This is a classical model of **Self-Organized Criticality (SOC)**. It exhibits:

- Emergent fractal patterns
- Scale-invariant avalanches
- Critical behaviour without parameter fine-tuning

For a deeper understanding of self-organized criticality in sandpile models, see [3].

8.3. The Sandpile Fractal

If the forcing term is given by

$$G(t, x, y) = \delta(x)\delta(y),$$

with initial condition

$$u(0, x, y) = 0,$$

and no boundary condition imposed, then the solution exhibits a fractal structure resembling the classical sandpile model. The resulting pattern looks as follows:

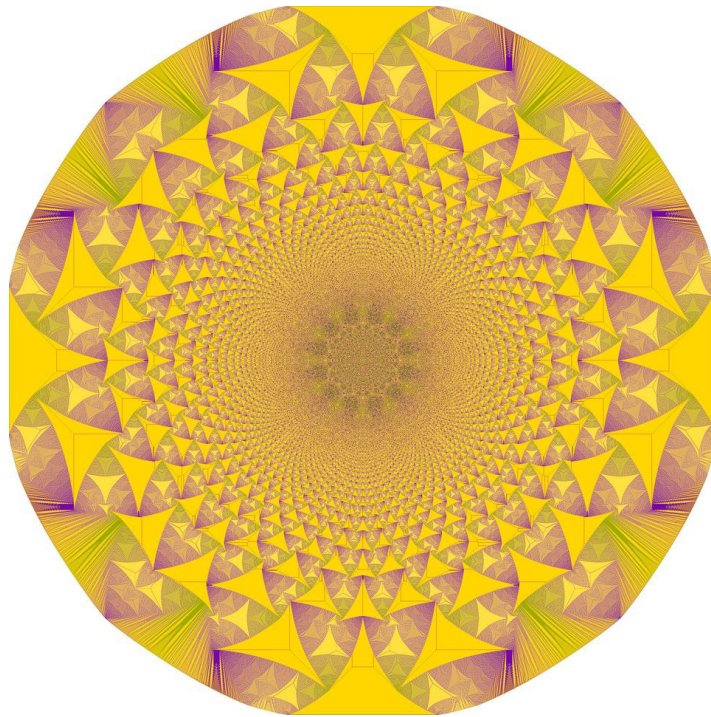


Figure 2. Fractal structure generated by dropping 30 million grains at a single site on the infinite square lattice and applying the rules of the Abelian sandpile model. Image by colt_browning, from Wikipedia, licensed under CC BY 4.0. Code: github.com/colt-browning/sandpile

8.4. Extended Sandpile Model

We present an extended version of the Abelian Sandpile Model in which each site topples only when it contains at least 8 grains of sand. The neighbourhood used here is the Moore neighbourhood, which includes the 8 surrounding cells (horizontal, vertical, and diagonal neighbours).

The evolution equation of the system is given by:

$$E_t u = u - 8\theta(u - 8) + \sum_{(i,j) \in M} \theta(E_x^i E_y^j u - 8) + G(t, x, y)$$

where:

- $u = u(t, x, y)$ represents the number of sand grains at position (x, y) and time t ,
- θ is the Heaviside step function,
- M denotes the Moore neighbourhood:
 $M = \{(-1, -1), (-1, 0), (-1, 1), (0, -1), (0, 1), (1, -1), (1, 0), (1, 1)\},$
- $G(t, x, y)$ is an external sand input term.

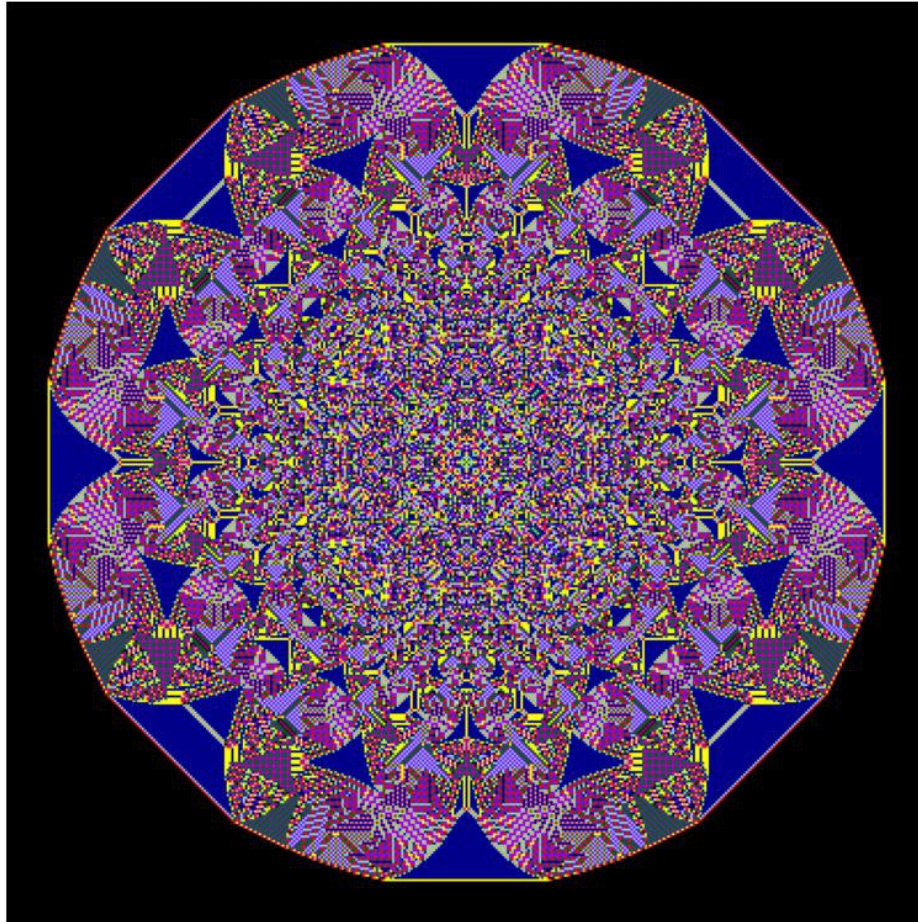
Example

- **Initial Condition:** $u(0, x, y) = 0$

- **Boundary Condition:** None (infinite or sufficiently large grid with no boundary constraints)
- **Forcing Term:** $G(t, x, y) = \delta(x)\delta(y)$ (unit sand grain dropped at the center)

The figure below shows the configuration $u(500000, x, y)$ after 500,000 iterations:

Sandpile Step 500000 (Topple every 500)



We observe that the resulting pattern resembles the classical sandpile model, but exhibits greater complexity due to the use of the Moore neighbourhood and the 8-grain threshold. The colours are more diverse, and the structure appears more intricate and vibrant.

8.5. Explanation for the Self-Organized Criticality

Consider the comparison between Rule 90 and the Sandpile model. Rule 90 is defined by the update rule:

$$E_t u = \text{mod}_2 \left(E_x^{-1} u + E_x u \right)$$

Both Rule 90 and the Sandpile model are based on local interactions and exhibit nonlinear dynamics. However, they display drastically different macroscopic behaviours: Rule 90 leads to global chaos, while the Sandpile model exhibits self-organized criticality.

In Rule 90, any small perturbation is exponentially amplified due to the absence of thresholds, every update directly sums the neighbouring values modulo 2, regardless of the current state. As a result, the system exhibits global sensitivity to initial conditions, a hallmark of chaos.

In contrast, the Sandpile model incorporates a threshold-based collapse mechanism. Before any site reaches the threshold (typically 4 in 2D), the system behaves linearly: each grain of sand simply increases the local state by 1. Once the threshold is reached, a toppling event occurs, distributing sand to neighbours and potentially triggering further topplings. This collapse phase introduces nonlinearity and chaotic propagation, but only in localized regions that have reached the critical state.

Therefore, the Sandpile model alternates between two distinct regimes:

- **Linear growth phase:** Sites far below the threshold evolve smoothly and predictably.
- **Chaotic collapse phase:** Sites at the threshold trigger avalanches with unpredictable, nonlinear dynamics.

Because the initial configuration is typically random, some sites are near-zero (linear regime), while others are near the threshold (nonlinear regime). This coexistence of order and chaos in space and time allows the Sandpile model to maintain a dynamically balanced state without the need for fine-tuning of parameters.

This explains why the Sandpile model self-organizes into a critical state: the rules themselves inherently support a mixture of linear and chaotic zones, allowing the system to spontaneously hover at the edge of chaos. Unlike systems that require tuning to a critical point, the Sandpile model exhibits critical behaviour as an emergent property of its local update rules.

8.6. Explanation for the Power Law Distribution

The avalanche size distribution in the Sandpile model follows a power law, typically of the form:

$$P(s) \propto s^{-\tau}$$

where $P(s)$ is the probability of an avalanche of size s , and τ is a constant exponent. This behaviour arises from the interplay between local dynamics, chaotic propagation, and external driving. We explain it in three parts:

1. **Why are small avalanches more common than large ones?**
Because the dynamics are based on local interactions, triggering a large avalanche requires a long chain of causally connected sites to be near the threshold, which is statistically less likely. In contrast, small avalanches require only a short chain and are thus much more probable.
2. **Why do large avalanches still occur frequently (not exponentially rare)?**
The collapse dynamics are chaotic: once a critical site topples, small perturbations can lead to unpredictable and explosive cascades, this is the butterfly effect. Therefore, even though rare, large avalanches are possible due to the system's inherent sensitivity at the critical state.
3. **Why is there no characteristic scale in avalanche size?**
The update rules impose no intrinsic limit on the spatial extent of an avalanche. The avalanche size depends entirely on the current configuration of the system. If a small localized region is near-critical, only a small avalanche occurs. But if a large domain is close to the threshold, a large avalanche may follow. Furthermore, the external driving is random and slow, allowing the system to evolve into configurations where large avalanches are statistically allowed. Thus, the system exhibits scale-invariant behaviour.

9. OFC Earthquake Model

9.1. OFC Earthquake Equation

The Olami-Feder-Christensen (OFC) Model [10] is a discrete-time cellular automaton designed to simulate earthquake dynamics using a self-organized criticality (SOC) framework. The model evolves according to the following partial difference equation:

$$E_t u = u - \theta(u - 1) + \alpha \sum_{(i,j) \in V} \theta(E_x^i E_y^j u - 1) + G(t, x, y)$$

where:

- $u = u(t, x, y) \in [0, 1]$ represents the local shear stress (or load) at site (x, y) at time t , $u : \mathbb{Z}^3 \rightarrow \mathbb{R}$.
- $E_t u = u(t + 1, x, y)$ denotes the stress at the next time step.

- $\theta(x)$ is the Heaviside function, defined as:

$$\theta(x) = \begin{cases} 1, & \text{if } x \geq 0 \\ 0, & \text{otherwise} \end{cases}$$

- $\alpha \sum_{(i,j) \in V} \theta(E_x^i E_y^j u - 1)$ represents the redistributed stress received from neighbouring sites $(i, j) \in V$ that have exceeded the failure threshold. α is a constant.
- $G(t, x, y)$ is an external driving term, typically very small, which represents slow tectonic loading. In most implementations, $G(t, x, y) = \varepsilon$ at a randomly selected site and zero elsewhere, modeling slow, local buildup of stress.

The stress u is usually constrained within the interval $[0, 1)$, with the threshold for failure set at $u = 1$. When $u \geq 1$, the site fails, redistributes stress to its neighbours, and resets or reduces its own stress, initiating potential cascades (earthquake avalanches) through the system.

This model is analogous to the sandpile model but includes continuous-valued stress and dissipation, making it more suitable for simulating earthquake-like events in tectonic systems.

9.2. Observations and Phenomena

The Olami–Feder–Christensen (OFC) earthquake model exhibits dynamics similar to the sandpile model. It demonstrates a power-law distribution in event sizes, where small avalanches occur frequently, while large avalanches are rare. This behaviour is a hallmark of *self-organized criticality* (SOC), indicating that the system naturally evolves toward a critical state without fine-tuning of parameters.

10. Forest Fire Model

10.1. Forest Fire Equation

We define the forest fire dynamics on a discrete lattice as a Partial Difference Equation with mod-3 state cycling [11]:

$$E_t u = \text{mod}_3(2 \delta(u - 1) \theta(S - 1) + G(t, x, y))$$

$$S = \sum_{(i,j) \in M} \delta(E_x^i E_y^j u - 2)$$

10.2. Definitions

- $u = u(t, x, y) \in \{0, 1, 2\}$ is the state of the cell at time t and spatial coordinates (x, y) .
 - 0: empty ground (bare land)
 - 1: tree
 - 2: burning tree
- E_t is the forward time shift operator: $E_t u := u(t + 1, x, y)$
- $E_x^i E_y^j u := u(t, x + i, y + j)$ are spatial shift operators in the Moore neighbourhood M .
- $\delta(n)$ is the Kronecker delta function: $\delta(n) = 1$ if $n = 0$, else 0.
- $\theta(x)$ is the Heaviside step function: $\theta(x) = 1$ if $x \geq 0$, else 0.
- $G(t, x, y) \in \{0, 1\}$ is the non-autonomous external driving force:
 - $G = 0$: no external change
 - $G = 1$: apply one-step growth/fire transition (see below)

10.3. Transition Logic (via mod 3)

The entire system evolves via modular arithmetic:

$$\text{New state} = \text{mod}_3(\text{current state} + \text{external or fire-induced increment})$$

- $0 \xrightarrow{+1} 1$: tree grows on empty land
- $1 \xrightarrow{+1} 2$: tree catches fire
- $2 \xrightarrow{+1} 0$: burning tree becomes ash (empty)

This simple cyclic structure captures the dynamics of vegetation growth, fire propagation, and destruction.

10.4. Interpretation

A tree at (x, y) will ignite if: - Its current state is 1 (tree), and - There is at least one burning tree in its neighbourhood ($S \geq 1$)

Otherwise, only external driving force $G(t, x, y)$ determines its evolution.

10.5. Observations and Phenomena

We consider the autonomous equation of the Forest Fire Model, and using Moore Neighbourhood. Numerical simulations reveal that the global behaviour of the system is highly sensitive to the initial condition, particularly the forest density.

Assume the initial condition where the entire leftmost column of the grid is set on fire, and all other cells are either trees or empty ground according to a prescribed forest density. We observe the following distinct behaviours:

- **Low density (e.g., 10%):** The fire fails to propagate. Due to the sparsity of trees, the flame quickly extinguishes after burning only a few connected trees.
- **High density (e.g., 80%):** The fire spreads rapidly and extensively. Almost the entire forest is consumed in a short period, with the flame front advancing in a nearly uniform and predictable wave.
- **Intermediate density (e.g., around 40%):** A highly nontrivial behaviour emerges. The fire propagates in an irregular, branching manner, forming a complex fractal-like structure. The fire path becomes winding, unpredictable, and sensitive to small variations in the initial configuration.

This suggests the existence of a critical threshold of forest density that separates the extinguishing, fractal, and total-burning regimes. The system exhibits characteristics of critical phenomena and self-organized complexity at intermediate densities.

11. Kuramoto Firefly Model

11.1. Kuramoto Equation

We define the Kuramoto model as a discrete phase-coupled oscillator network evolving on a 2D grid:

$$E_t u = \text{mod}_{2\pi} \left(u + w(x, y) + \frac{K}{8} \sum_{(i,j) \in M} \sin(E_x^i E_y^j u - u) \right)$$

- $u = u(t, x, y) \in [0, 2\pi]$ is the phase at grid point (x, y) and time t .
- $w(x, y)$ is a fixed intrinsic frequency assigned randomly to each point.
- K is the coupling constant. In our simulations, we choose $K = 1$.
- M is the Moore neighbourhood:

$$M = \{(\pm 1, 0), (0, \pm 1), (\pm 1, \pm 1)\}$$

11.2. Principle of Locality

In the original Kuramoto model, oscillators are globally coupled, each unit interacts with every other oscillator in the system. [12]. While this assumption simplifies mathematical analysis, it is biologically implausible. For example, it is unreasonable to assume that a firefly can perceive and compute the phases of all other fireflies in the environment at each time step.

To address this limitation, we propose the **Kuramoto Firefly Model**, which enforces the **Principle of Locality**. In this model, each oscillator interacts only with its immediate neighbours (e.g., a Moore neighbourhood in two dimensions). This localized interaction is more consistent with real world biological systems and gives rise to rich emergent phenomena such as synchronization, topological defects, and vortex-like phase structures.

11.3. Observations and Phenomena

In the simulation, we use periodic boundary condition.

From the spatiotemporal plots of the system, we observe the following:

- The system exhibits **topological defects**: points around which the phase (colour) continuously wraps from 0 to 2π .
- These defects appear in **pairs**, with opposite rotational direction:
 - One clockwise
 - One counterclockwise
- When these defects move and eventually **annihilate** each other, a large region of **synchronization** emerges suddenly.

This model shows interesting connections to:

- **Phase synchronization phenomena**
- **Topological vortex dynamics**
- **Pattern formation and self-organization**

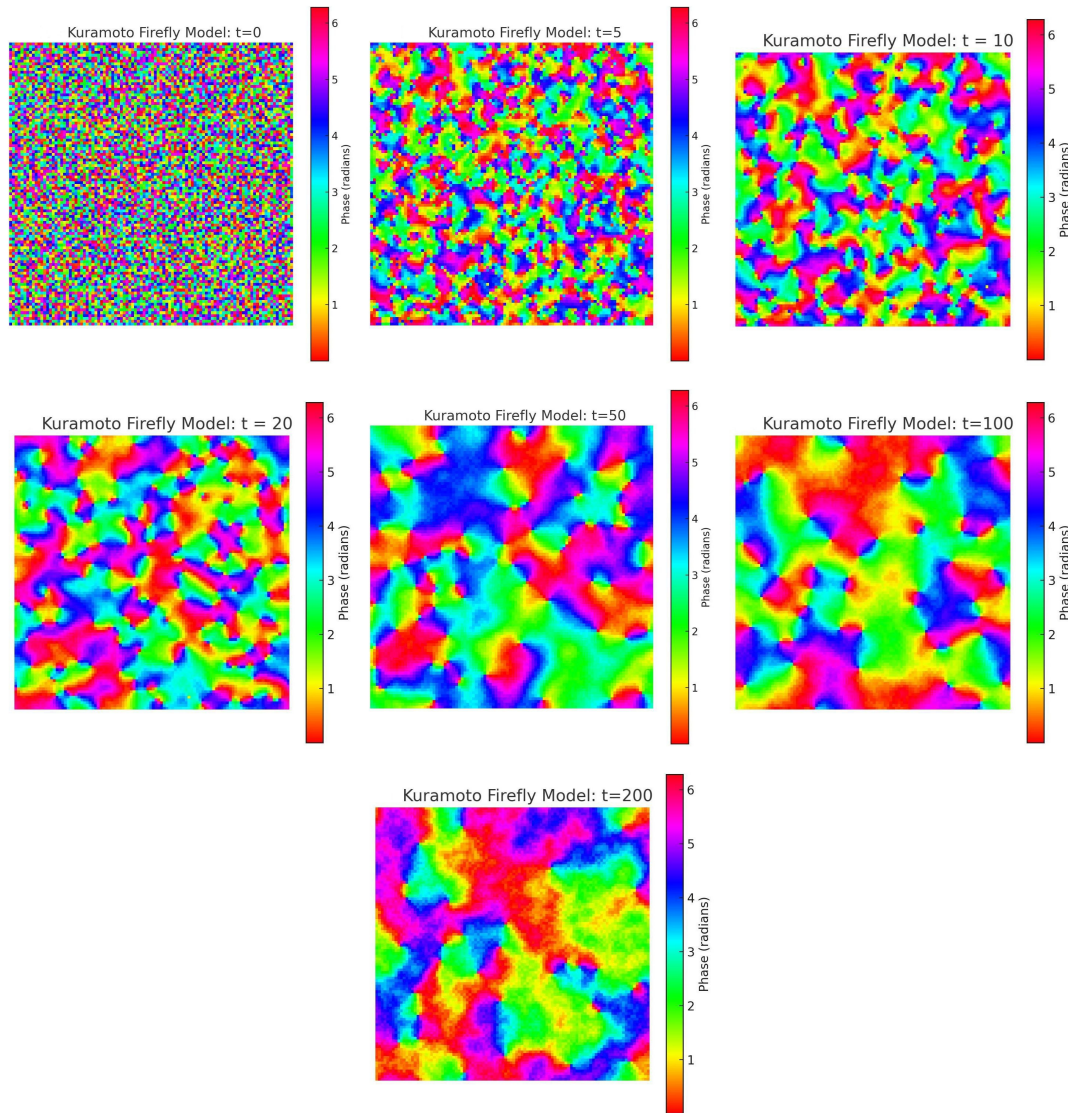


Figure 12. Spatiotemporal evolution of the Kuramoto model. Topological defects (vortices) emerge and move, and their annihilation leads to large-scale synchronization.

12. Ising Model

12.1. Governing Equation

We propose the following discrete evolution equation for the well-known Ising Model [13]:

$$E_t u = (1 - \delta(\text{sign}(JS_V u) - \text{sign}(u))) \theta(F - \varepsilon) \text{sign}(JS_V u) + \delta(\text{sign}(JS_V u) - \text{sign}(u)) \theta(F - F_0) (-\text{sign}(JS_V u)) \quad (1)$$

where

$$S_V u = \sum_{(i,j) \in V} E_x^i E_y^j u,$$

- V is the von Neumann neighbourhood.
- $u = u(t, x, y) \in \{-1, +1\}$ is the binary state at discrete time t and spatial coordinate (x, y) .
- $E_t u := u(t + 1, x, y)$ is the next time step value.
- $F = F(t, x, y) \sim \mathcal{N}(\mu, c^2)$ is a random driving field at site (x, y) , drawn from a normal distribution centered at $\mu = kT$, where T is the temperature and k is a constant.

- ε is a small positive number (baseline activation threshold).
- F_0 is a larger threshold required for flipping a stable site.
- J is the coupling constant determining interaction strength between neighbouring sites.

12.2. Physical Interpretation

The function $u(t, x, y)$ represents the spin (or state) of a particle or site at position (x, y) and time t . The variable $E_t u$ denotes the updated state at time $t + 1$. The update is governed by local interactions and stochastic environmental drive F .

- The **first term** activates when the site's current state is *different* from the local neighbourhood majority ($\text{sign}(J S_V u) \neq \text{sign}(u)$), allowing it to flip easily when $F > \varepsilon$.
- The **second term** activates when the site's current state *matches* the neighbourhood majority, but may still flip if the driving force exceeds a higher threshold F_0 , modeling thermal noise or instability.

This model expresses the Ising-like behaviour in the language of partial difference equations (PΔE), linking local deterministic update rules with stochastic driving force under thermal control.

13. Greenberg–Hastings Model

13.1. Greenberg–Hastings Equation

We propose the following formulation of the Greenberg–Hastings model [14] as a partial difference equation (PΔE):

$$E_t u = \theta(u - 1) \cdot \text{mod}_n(u + 1) + \delta(u) \cdot \theta(S - b)$$

where

$$S = \sum_{(i,j) \in N} \delta(E_x^i E_y^j u - 1)$$

with the following definitions:

- $u = u(t, x, y) \in \mathbb{Z}$, a discrete function defined on space-time: $u : \mathbb{Z}^3 \rightarrow \mathbb{Z}$.
- $\theta(x) = \begin{cases} 1, & x \geq 0 \\ 0, & x < 0 \end{cases}$ is the Heaviside step function.
- $\delta(x) = \begin{cases} 1, & x = 0 \\ 0, & x \neq 0 \end{cases}$ is the Kronecker delta function.
- $\text{mod}_n(x) = x - n \cdot \lfloor \frac{x}{n} \rfloor$ is the modulo- n operation.
- $n \in \mathbb{N}$ is the number of discrete states (typically $n \geq 3$).
- $b \in \mathbb{N}$ is the excitation threshold.
- N is the set of relative spatial coordinates in the neighbourhood (e.g., von Neumann or Moore).

13.2. Explanation

The Greenberg–Hastings model is a classical excitable medium model capable of producing complex spatiotemporal patterns such as spiral waves. Traditionally, it is expressed using state variables $x_{i,j}^n$ updated via nested if-else rules.

In this work, we reformulate the model as a partial difference equation (PΔE) using evolution operators and logical functions. This approach provides a more elegant, symbolic, and generalizable representation that integrates naturally into our proposed framework of discrete dynamical systems.

14. The Langton's Ant

14.1. The Langton's Ant Equations

We describe the classical Langton's Ant [15] using a system of coupled ordinary and partial difference equations.

The update rules are given as follows:

$$\begin{aligned} u(t+1, x, y) &= u(t, x, y) + (1 - 2u(t, x, y)) \cdot \delta(x - X(t))\delta(y - Y(t)) \\ d(t+1) &= (d(t) + (1 - 2u(t, X(t), Y(t)))) \bmod 4 \\ X(t+1) &= X(t) + \cos\left(\frac{\pi}{2} \cdot d(t+1)\right) \\ Y(t+1) &= Y(t) - \sin\left(\frac{\pi}{2} \cdot d(t+1)\right) \end{aligned}$$

Here, $\delta(\cdot)$ denotes the Kronecker delta function:

$$\delta(x) = \begin{cases} 1 & \text{if } x = 0 \\ 0 & \text{otherwise} \end{cases}$$

The above equations can be rewritten in operator form as:

$$\begin{aligned} E_t u &= u + (1 - 2u) \cdot \delta(x - X)\delta(y - Y) \\ E_t d &= \text{mod}_4(d + (1 - 2u(t, X, Y))) \\ E_t X &= X + \cos\left(\frac{\pi}{2} \cdot E_t d\right) \\ E_t Y &= Y - \sin\left(\frac{\pi}{2} \cdot E_t d\right) \end{aligned}$$

where:

- $u = u(t, x, y) \in \{0, 1\}$ is the state of the lattice at time t and position (x, y) , i.e., $u : \mathbb{Z}^3 \rightarrow \{0, 1\}$.
- $d = d(t) \in \{0, 1, 2, 3\}$ is the direction of the ant at time t , where $d : \mathbb{Z} \rightarrow \mathbb{Z}_4$.
- $(X(t), Y(t)) \in \mathbb{Z}^2$ is the location of the ant, with $X, Y : \mathbb{Z} \rightarrow \mathbb{Z}$.
- The modulo function is defined as: $\text{mod}_4(x) = x - 4 \cdot \lfloor \frac{x}{4} \rfloor$.

This is a system of coupled difference equations consisting of:

- One partial difference equation for the lattice state $u(t, x, y)$.
- Three ordinary difference equations for the ant's direction and position: $d(t)$, $X(t)$, and $Y(t)$.

14.2. Observations and Phenomena

The following figure illustrates the evolution of Langton's Ant over discrete time steps. The system exhibits **transient chaos**, followed by the emergence of a repeating structure called the *highway*, which acts as a **periodic attractor** in the phase space.

15. The Discrete Parallel Universe of PDE

15.1. Motivation

In the course of formulating a wide class of complex systems using Partial Difference Equations (PΔE), we observed that their resemblance to Partial Differential Equations (PDE) goes far beyond the multivariable function structure.

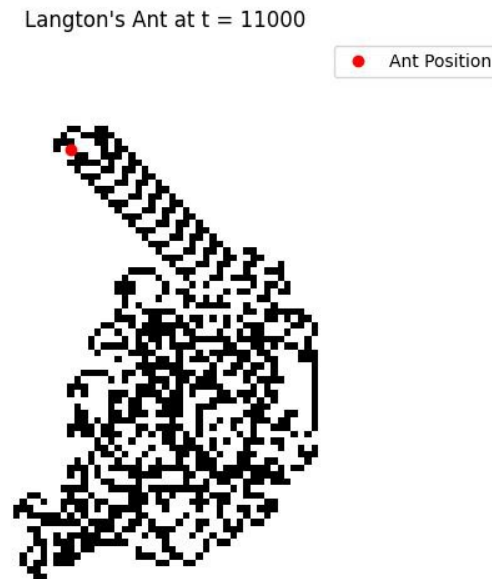


Figure 13. The evolution of Langton's Ant over time. The black squares represent flipped cells (state 1), the white background denotes unvisited cells (state 0), and the red dot indicates the current position of the ant at the final time step.

Specifically, we found deep analogies between the two worlds in terms of topology, geometry, symmetry, function spaces, algebraic structure, and even connections to number theory and combinatorics.

These similarities are not superficial; they suggest a profound **duality** between the continuous and the discrete, a kind of "mirror universe" where the traditional PDE-based framework has a structurally equivalent counterpart in the discrete domain.

This section aims to articulate and formalize these analogies by providing a systematic comparison between PDE and PΔE across multiple mathematical domains.

15.2. Calculus and Discrete Calculus

In the theory of **Partial Differential Equations (PDE)**, the function u is typically a continuous multivariable function:

$$u : \mathbb{R}^n \rightarrow \mathbb{C}$$

In PDE, we use the differential operator, such as the partial derivative:

$$\frac{\partial u}{\partial t}$$

We also use integration to accumulate values over continuous domains:

$$\int_a^b f(x) dx$$

In contrast, in the theory of **Partial Difference Equations (PΔE)**, the function u is a discrete multivariable function:

$$u : \mathbb{Z}^n \rightarrow \mathbb{C}$$

We use difference operators [16], such as:

$$\Delta_t u = E_t u - u \quad \text{and shift operators} \quad E_t u = u(t+1, x, \dots)$$

Instead of integrals, we use summation over discrete domains:

$$\sum_{x=a}^b f(x)$$

This comparison highlights the foundational parallel between classical calculus and discrete calculus, which provides the analytical tools for studying dynamics on discrete domains such as lattices or graphs.

15.3. ODE and OΔE

When we focus solely on the evolution of a single point over time, neglecting spatial structure, the continuous formulation reduces to an **Ordinary Differential Equation (ODE)**:

$$\frac{du}{dt} = f(u, t), \quad u : \mathbb{R} \rightarrow \mathbb{C}$$

In the discrete formulation, this becomes an **Ordinary Difference Equation (OΔE)**:

$$E_t u = f(u, t), \quad u : \mathbb{Z} \rightarrow \mathbb{C}, \quad E_t u := u(t+1)$$

A Partial Differential Equation (PDE) can be interpreted as an infinite system of coupled ODEs, where each spatial point evolves in time according to both its own value and its neighbours'.

Likewise, when boundary conditions are ignored, a Partial Difference Equation (PΔE) corresponds to an infinite system of coupled OΔEs, one for each spatial site on the discrete lattice.

This perspective emphasizes the structural parallelism between the continuous and discrete dynamical systems, revealing how local update rules (whether differential or difference-based) can form complex global behaviour through coupling.

15.4. Initial and Boundary Conditions

In the theory of **Partial Differential Equations (PDE)**, the initial condition is typically given as a continuous function. For example:

$$u(0, x) = f(x), \quad f \in C^1(\mathbb{R})$$

The boundary [6] of the spatial domain is usually a smooth curve, surface, or differentiable manifold. Boundary conditions can be classified into:

- **Dirichlet condition:** Specifies the value of the function at the boundary.
- **Neumann condition:** Specifies the value of the normal derivative at the boundary.
- **Robin condition:** Specifies a linear combination of the function and its derivative at the boundary.

In contrast, for **Partial Difference Equations (PΔE)**, the initial condition is defined as a discrete function. For instance:

$$u(0, x) = f(x), \quad f : \mathbb{Z} \rightarrow \mathbb{C}$$

The spatial domain is defined over a lattice or graph structure, and the boundary may be a polygon (in 2D), a polyhedron (in 3D), or a higher-dimensional network.

PΔE can also incorporate discrete analogues of classical boundary conditions:

- **Discrete Dirichlet condition:** Specifies function values on the boundary nodes.
- **Discrete Neumann condition:** Specifies discrete gradients (differences) at the boundary.
- **Discrete Robin condition:** Specifies a linear relation between function value and difference at the boundary.

This parallel between continuous and discrete formulations illustrates the structural correspondence between PDE and PΔE in the treatment of initial and boundary data.

15.5. Laplace Transform and Z Transform

In the theory of continuous differential equations, the **Laplace transform** [17] is a powerful tool for solving linear ordinary or partial differential equations, particularly with constant coefficients and initial conditions. It is defined as:

$$\mathcal{L}[f(t)] = F(s) = \int_0^{\infty} e^{-st} f(t) dt$$

where $f(t)$ is a continuous function defined on $[0, \infty)$, and $s \in \mathbb{C}$ is the complex frequency parameter. The Laplace transform converts a differential equation into an algebraic equation in the s -domain, which can then be solved more easily.

In contrast, in the discrete setting of difference equations, the corresponding transform is the **Z-transform** [5], which is used to solve linear recurrence relations and difference equations. It is defined as:

$$\mathcal{Z}[f(n)] = F(z) = \sum_{n=0}^{\infty} f(n) z^{-n}$$

where $f(n)$ is a discrete function defined on $\mathbb{Z}_{\geq 0}$, and $z \in \mathbb{C}$ is the complex variable. Like the Laplace transform, the Z-transform converts difference equations into algebraic equations in the z -domain.

Both transforms play analogous roles in their respective settings: converting dynamical equations into algebraic equations, enabling the application of tools from complex analysis, control theory, and systems theory.

15.6. Fourier Transform and Discrete Fourier Transform

In the theory of partial differential equations (PDE), **Fourier series** and the **Fourier transform** [6] are fundamental tools used to analyze and solve linear PDE, especially with periodic or infinite domain conditions. These methods decompose functions into sums or integrals of sinusoidal components, revealing their frequency content.

Fourier Series (Continuous, Periodic)

For a function $f(x)$ defined on a finite interval $[0, L]$, the Fourier series is given by:

$$f(x) = \sum_{n=-\infty}^{\infty} \hat{f}_n e^{i \frac{2\pi n}{L} x} \quad \text{where} \quad \hat{f}_n = \frac{1}{L} \int_0^L f(x) e^{-i \frac{2\pi n}{L} x} dx$$

Fourier Transform (Continuous, Aperiodic)

For a non-periodic function $f(x) \in L^2(\mathbb{R})$, the Fourier transform is defined as:

$$\hat{f}(\xi) = \int_{-\infty}^{\infty} f(x) e^{-2\pi i \xi x} dx \quad \text{with inverse} \quad f(x) = \int_{-\infty}^{\infty} \hat{f}(\xi) e^{2\pi i \xi x} d\xi$$

These tools convert differential operators into algebraic ones in the frequency domain, enabling elegant solutions to linear PDE.

In the discrete setting of partial difference equations (PΔE), we have analogous tools: the **Discrete Fourier Series** (DFS) and the **Discrete Fourier Transform** (DFT) [18], which serve the same purpose in analyzing spatially discrete, periodic functions.

Discrete Fourier Series

Let $f[n]$ be a periodic discrete function defined on $n = 0, 1, \dots, N - 1$, then:

$$f[n] = \sum_{k=0}^{N-1} \hat{f}[k] e^{2\pi i k n / N} \quad \text{where} \quad \hat{f}[k] = \frac{1}{N} \sum_{n=0}^{N-1} f[n] e^{-2\pi i k n / N}$$

Discrete Fourier Transform (DFT)

The DFT is the same as DFS in form but used for numerical computation on finite-length signals:

$$\mathcal{F}[f]_k = \sum_{n=0}^{N-1} f[n] e^{-2\pi i k n / N} \quad \text{with inverse} \quad f[n] = \frac{1}{N} \sum_{k=0}^{N-1} \mathcal{F}[f]_k e^{2\pi i k n / N}$$

These transforms diagonalize linear, shift-invariant operators and are especially useful in solving linear PΔE with constant coefficients and periodic boundary conditions.

15.7. Discrete and Continuous Green's Function

For simplicity, we only consider the one-dimensional case.

Let $u = u(t, x)$, where $u : \mathbb{R}^2 \rightarrow \mathbb{R}$, and consider a linear partial differential equation (PDE) of the form

$$L(u) = 0,$$

where L is a linear operator. Suppose the initial condition is given by

$$u(0, x) = \delta(x),$$

where $\delta(x)$ is the Dirac delta function, and assume there are no boundary conditions. If the solution to this PDE is denoted by $G(t, x)$, the Green's function [6], then the solution corresponding to an arbitrary initial condition $u(0, x) = f(x)$ is given by the convolution:

$$u(t, x) = \int_{-\infty}^{\infty} f(s) G(t, x - s) ds.$$

Now, let $u = u(t, x)$, where $u : \mathbb{Z}^2 \rightarrow \mathbb{R}$, and consider a linear partial difference equation (PΔE) of the form

$$L(u) = 0,$$

where L is again a linear operator. Suppose the initial condition is

$$u(0, x) = \delta(x),$$

where $\delta(x)$ is now the Kronecker delta function, and assume there are no boundary conditions. If the solution is denoted by $G(t, x)$, the discrete Green's function, then for an arbitrary initial condition $u(0, x) = f(x)$, the solution is given by the discrete convolution:

$$u(t, x) = \sum_{s \in \mathbb{Z}} f(s) G(t, x - s).$$

In the continuous setting, the Green's function representation of the solution involves an integral over the spatial domain. In the discrete setting, the corresponding representation is given by a

summation. These two formulations are structurally analogous, differing only in the underlying measure, Lebesgue measure for the continuous case, and counting measure for the discrete case.

15.8. Gaussian Distribution and Combinatorics

In the continuous heat equation, the equation is given by

$$\frac{\partial u}{\partial t} = \alpha \frac{\partial^2 u}{\partial x^2}.$$

Consider the initial condition

$$u(0, x) = \delta(x),$$

where $\delta(x)$ is the Dirac delta function. With no boundary condition, the solution is

$$u(t, x) = \frac{1}{\sqrt{4\pi\alpha t}} \exp\left(-\frac{x^2}{4\alpha t}\right),$$

which is the Gaussian distribution.

In the discrete heat equation, the equation becomes

$$\Delta_t u = \alpha \nabla^2 u,$$

and choosing $\alpha = \frac{1}{2}$, we write the equation as

$$E_t u = \frac{1}{2} (E_x u + E_x^{-1} u).$$

Consider the initial condition

$$u(0, x) = \delta(x),$$

where $\delta(x)$ is the Kronecker delta function, and again, no boundary condition is imposed.

Then the solution is given by

$$u(t, x) = \delta(\text{mod}_2(x+t)) \cdot C\left(t, \frac{x+t}{2}\right) \cdot \left(\frac{1}{2}\right)^t,$$

where

$$C(n, k) = \begin{cases} \frac{n!}{k!(n-k)!} & \text{if } 0 \leq k \leq n, \\ 0 & \text{otherwise.} \end{cases}$$

We can see that the Gaussian distribution is closely related to combinatorics. In fact, the Gaussian distribution emerges as the continuous limit of the binomial distribution as the number of trials tends to infinity, with appropriate scaling and centering, a consequence of the Central Limit Theorem.

15.9. Discrete and Continuous Dynamical Systems

In the framework of topological dynamics, a **dynamical system** is defined as a pair (X, φ^t) , where X is a topological space called the *state space*, and $\varphi^t : X \rightarrow X$ is a family of time-evolution operators indexed by $t \in T$, where $T = \mathbb{R}$ (continuous time) or $T = \mathbb{Z}$ (discrete time). The map φ^t satisfies the semigroup property $\varphi^{t+s} = \varphi^t \circ \varphi^s$ and $\varphi^0 = \text{id}$ [19].

Continuous Dynamical Systems.

Partial Differential Equations (PDE) define continuous dynamical systems, typically of the form:

$$u : \mathbb{R}^n \rightarrow \mathbb{C}, \quad \text{e.g., } u(t, x, y, z), \quad \text{with } t \in \mathbb{R}, (x, y, z) \in \Omega \subseteq \mathbb{R}^3.$$

The state space X is a continuous function space, such as $L^2(\Omega)$, $C^k(\Omega)$, or Sobolev spaces $H^k(\Omega)$. The evolution of the system is governed by differential operators that act on these function spaces. In such systems, chaotic behaviour is rare and usually manifests as smooth strange attractors—continuous curves resembling vortices or spirals. The attractors tend to be geometrically regular and topologically simple.

Discrete Dynamical Systems.

Partial Difference Equations (PΔE) define discrete dynamical systems, where

$$u : \mathbb{Z}^n \rightarrow \mathbb{C}, \quad \text{e.g., } u(t, x, y, z), \quad \text{with } t \in \mathbb{Z}, (x, y, z) \in \Omega \subseteq \mathbb{Z}^3.$$

The state space X is a *discrete function space*, i.e., a space of functions defined on a lattice, often taking values in \mathbb{C} , \mathbb{Z} , or even a finite set (such as Boolean values), for example, discrete Hilbert Space: $\mathcal{L}^2(\Omega)$. Such systems are capable of generating highly complex spatiotemporal patterns and exhibit rich chaotic behaviours. The associated strange attractors can have fractal geometries far more exotic than in continuous systems: they may resemble turbulent flows, ribbons, spider webs, tunnels, wings, bottles, or other intricate structures.

Fractals and Complexity.

Fractals are rarely seen in PDE due to the smoothness constraints imposed by the function space and differential operators. In contrast, PΔE can easily generate fractal structures. For example:

- Rule 90 cellular automaton generates the Sierpiński triangle.
- The Abelian Sandpile Model exhibits discrete self-organized criticality with fractal avalanche clusters.
- Coupled Map Lattices generate chaotic spatial patterns with complex attractors.

These observations suggest that discrete dynamical systems possess a fundamentally different character compared to continuous ones: their state spaces are inherently combinatorial, their attractors more irregular, and their emergent behaviours often more computationally universal and structurally complex.

15.10. Manifolds and Networks

Partial Differential Equations (PDE) are traditionally defined on smooth manifolds [20], such as curves, surfaces, or general Riemannian manifolds. The underlying space is continuous, and thus the study of PDE often requires tools from differential geometry. PDE can be formulated in various coordinate systems, including rectangular, cylindrical, and spherical coordinates, depending on the geometry of the manifold.

In contrast, we observe that Partial Difference Equations (PΔE) need not be confined to rectangular grids. They can also be defined on triangular lattices, hexagonal lattices, and other non-rectangular discretizations [21]. Furthermore, PΔE can be formulated on polyhedral surfaces such as Goldberg polyhedra, zonohedra, and even higher-dimensional polytopes [22]. In this case, the underlying structure is a discrete network, and the evolution takes place over this network.

The study of such discrete geometric networks naturally connects to graph theory and discrete differential geometry. Moreover, PΔE exhibit deep relationships with discrete and computational geometry.

Although PDE and PΔE belong to continuous and discrete domains respectively, the geometric structures underlying them exhibit a fascinating duality. This suggests a possible formal correspondence between smooth manifolds and discrete networks, and opens a path to unify the continuous and discrete geometric theories under a common framework.

15.11. Boolean Algebra, Theoretical Computer Science and Number Theory

In certain classes of Partial Difference Equations (PΔE), such as those arising from elementary cellular automata, the state space is restricted to two values, typically $\{0, 1\}$. This naturally connects such systems to Boolean algebra, where logical operations govern the update rules. The evolution of these discrete systems is thus governed by logical expressions, and the resulting dynamics can often be described entirely in terms of Boolean functions.

Moreover, the well-known *Conway's Game of Life*, which can also be viewed as a particular PΔE defined over a two-dimensional grid with binary states, is known to be Turing complete [23]. This highlights a deep and important connection between PΔE and theoretical computer science [24], as such systems are capable of universal computation. Cellular automata and related PΔE systems therefore serve as bridges between dynamical systems, logic, and computation.

Additionally, difference equations have strong ties to number theory [24]. For example, the Fibonacci sequence satisfies the recurrence relation

$$x(t+1) = x(t) + x(t-1),$$

which is a second-order linear ordinary difference equation. More generally, integer-valued solutions to linear PΔE often exhibit rich combinatorial and arithmetic structures. In many cases, the solutions involve binomial coefficients, discrete convolutions, or integer sequences catalogued in number theory. The fact that PΔE naturally accommodate integer-valued functions $u(t, x) \in \mathbb{Z}$ reinforces their relevance to number-theoretic investigations.

Hence, PΔE lie at the intersection of Boolean algebra, number theory, and computational theory, offering a unified framework for exploring discrete structures and their evolution.

15.12. Summary

We observe that although partial differential equations (PDE) and partial difference equations (PΔE) differ fundamentally in nature, one being continuous and the other discrete, they exhibit profound structural similarities, often appearing as duals of one another.

Moreover, we have seen that PΔE are deeply interconnected with various domains of discrete mathematics, including combinatorics, number theory, graph theory, and discrete and computational geometry.

Partial Difference Equations are not merely exotic multivariable recurrence relations; rather, they serve as a unifying framework that bridges numerous areas in the discrete mathematical landscape.

16. Summary

16.1. From Numerical Methods to Language of Complexity

We began this work by introducing *Partial Difference Equations* (PΔE) as a formal mathematical framework for modeling systems that are discrete in both time and space. Starting from their motivation and definitions, we established their relationship to ordinary difference equations (OΔE) and to their continuous counterparts, partial differential equations (PDE), thereby clarifying the scope and intent of this study.

We then examined the linear theory, presenting the main types of linear PΔE and illustrating simple analytic solutions without pursuing a full solution theory. This led naturally to the development of discrete functional analysis tools: discrete function spaces, Hilbert space structures, and discrete operators, including shift and difference operators, along with their adjoint properties. Within this setting, we defined the discrete Green's function and demonstrated how it parallels its continuous analogue.

The unifying strength of the PΔE formalism was then illustrated by reformulating a broad spectrum of well-known discrete models, ranging from cellular automata and coupled map lattices to sandpile models, the Ising model, and beyond, into explicit PΔE form. These examples demonstrate

that systems traditionally studied in isolation can be expressed within a single coherent mathematical language.

Finally, we compared the “discrete universe” of PΔE with the continuous universe of PDE, revealing deep structural parallels while emphasizing their distinct mathematical foundations. In this view, PΔE and PDE emerge as mathematical twins: discrete–continuous analogues that share a common formal architecture yet belong to fundamentally different domains of mathematics.

From their origins as tools in numerical methods, difference equations have evolved here into a general *language of complexity*, capable of describing, unifying, and extending a wide variety of models in physics, biology, and computation. This perspective not only enriches the theoretical foundations of discrete dynamical systems but also opens pathways for future research across the sciences.

References

1. Murray, J. *Mathematical Biology: I. An Introduction*, third ed.; Vol. 17, *Interdisciplinary Applied Mathematics*, Springer-Verlag: New York, Berlin, Heidelberg, 2002. Printed in the United States of America.
2. Murray, J. *Mathematical Biology II: Spatial Models and Biomedical Applications*, third ed.; Vol. 18, *Interdisciplinary Applied Mathematics*, Springer-Verlag: New York, Berlin, Heidelberg, 2003. Printed in the United States of America.
3. Bak, P.; Tang, C.; Wiesenfeld, K. Self-Organized Criticality: An Explanation of $1/f$ Noise. *Physical Review Letters* **1987**, *59*, 381–384.
4. Johnston, N.; Greene, D. *Conway’s Game of Life: Mathematics and Construction*; Self-published, 2022; p. 492. Hardcover, colour printing; US letter size (8.5 × 11 in), <https://doi.org/10.5281/zenodo.6097284>.
5. Elaydi, S. *An Introduction to Difference Equations*, 3rd ed.; Undergraduate Texts in Mathematics, Springer: New York, 2005.
6. Olver, P.J. *Introduction to Partial Differential Equations*; Undergraduate Texts in Mathematics, Springer: Cham, Heidelberg, New York, Dordrecht, London, 2014. <https://doi.org/10.1007/978-3-319-02099-0>.
7. Wolfram, S. *A New Kind of Science*, 1st ed.; Wolfram Media: Champaign, Illinois, 2002.
8. Itoh, M.; Chua, L.O. Difference Equations for Cellular Automata. *International Journal of Bifurcation and Chaos* **2009**, *19*, 805–830.
9. Yanagita, T.; Kaneko, K. Coupled map lattice model for convection. *Physics Letters A* **1993**, *175*, 415–420. [https://doi.org/10.1016/0375-9601\(93\)90993-A](https://doi.org/10.1016/0375-9601(93)90993-A).
10. Li, B.Q.; Wang, S.J. Self-Organized Criticality in an Anisotropic Earthquake Model. *Communications in Theoretical Physics* **2018**, *69*, 280–284. <https://doi.org/10.1088/0253-6102/69/3/280>.
11. Sun, X.; Li, N.; Chen, D.; Chen, G.; Sun, C.; Shi, M.; Gao, X.; Wang, K.; Hezam, I.M. A Forest Fire Prediction Model Based on Cellular Automata and Machine Learning. *IEEE Access* **2024**, *12*, 55389–55403. CC BY-NC-ND 4.0, <https://doi.org/10.1109/ACCESS.2024.3389035>.
12. Strogatz, S.H. From Kuramoto to Crawford: Exploring the Onset of Synchronization in Populations of Coupled Oscillators. *Physica D: Nonlinear Phenomena* **2000**, *143*, 1–20.
13. Singh, S.P. The Ising Model: Brief Introduction and Its Application. In *Solid State Physics*; IntechOpen, 2020. CC BY 3.0 license, <https://doi.org/10.5772/intechopen.90875>.
14. Wu, A.C.; Xu, X.J.; Wang, Y.H. Excitable Greenberg-Hastings cellular automaton model on scale-free networks. *arXiv preprint cond-mat/0701248* **2007**. arXiv:cond-mat/0701248, submitted 11 Jan 2007.
15. Boon, J.P. Langton’s Ant as an Elementary Turing Machine. In *Nonequilibrium Thermodynamics and Fluctuation Kinetics*; Springer, Cham, 2022; Vol. 208, *Fundamental Theories of Physics*, pp. 135–140. https://doi.org/10.1007/978-3-031-04458-8_7.
16. Mariconda, C.; Tonolo, A. *Discrete Calculus: Methods for Counting*; Vol. 103, *UNITEXT - La Matematica per il 3+2*, Springer International Publishing: Switzerland, 2016. <https://doi.org/10.1007/978-3-319-03038-8>.
17. Adkins, W.A.; Davidson, M.G. *Ordinary Differential Equations*; Undergraduate Texts in Mathematics, Springer, 2012.
18. Mutagi, R.N. Understanding the Discrete Fourier Transform. PDF lecture notes, January 2004. Uploaded by the author on ResearchGate, Indus University.
19. Brin, M.; Stuck, G. *Introduction to Dynamical Systems*, 1st ed.; Cambridge University Press, 2002. Graduate-level introduction to dynamical systems theory; Cambridge, 1st ed.
20. Petersen, P. *Riemannian Geometry*, 3rd ed.; Vol. 171, *Graduate Texts in Mathematics*, Springer, Cham, 2016. <https://doi.org/10.1007/978-3-319-26654-1>.

21. Devadoss, S.L.; O'Rourke, J. *Discrete and Computational Geometry*, 1st ed.; Princeton University Press, 2011.
22. Goodman, J.E.; O'Rourke, J.; Tóth, C.D., Eds. *Handbook of Discrete and Computational Geometry*, 3rd ed.; Discrete Mathematics and Its Applications, Chapman & Hall/CRC: Boca Raton, FL, 2017; pp. xix + 1,950. <https://doi.org/10.1201/9781315119601>.
23. Rendell, P. Game of Life Turing Machine. In *Turing Machine Universality of the Game of Life*; Adamatzky, A.; Martínez, G.J., Eds.; Springer International Publishing, Cham, 2016; Vol. 18, *Emergence, Complexity and Computation*, pp. 45–70. https://doi.org/10.1007/978-3-319-19842-2_4.
24. Rosen, K.H., Ed. *Handbook of Discrete and Combinatorial Mathematics*, 2nd ed.; Discrete Mathematics and Its Applications, Chapman & Hall/CRC, 2017; p. 1612. <https://doi.org/10.1201/9781315156484>.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.