Phenomenology of Tensor Modulation in Elementary Kinetic Automata

Yuri V. Shalygo
Gamma Ltd.
19, Nekrasova str.
Vyborg, 188800, Russia

This paper continues and extends the earlier works by the author on a novel model of a complex dynamical system called a kinetic automaton. The primary goal of the paper is to introduce an alternative tensor-based method of modulation and to demonstrate that it not only significantly enhances the tunability of the model and the complexity of its behavior, but also is able to emulate many other discrete-time continuous-state dynamical systems. The paper provides the results of the investigation of spatio-temporal patterns arising under different modes or parameters of modulation in elementary (one-dimensional) kinetic automata. Special attention is given to quantity conservation, which is the most salient feature of the model.

1. Introduction

This paper continues and extends the earlier works by the author on a novel model of a complex dynamical system called a *kinetic automaton*, or *kinon* for short.

The first paper [1] introduced a novel numerical algorithm, called conservative rank transform (CRT), in which not quantities as such but their relative values (ranks) are transformed, and the total quantity does not change after transformation. This method has become a guiding principle and computational kernel for the introduced kinon model, defined as a *reflexive* dynamical system with *active* transport. The model can be considered as the generalization of the lattice Boltzmann model (LBM) [2], which is not restricted to the Boltzmann equation and a regular grid. The key element of the model, making its properties and dynamics different from the LBM, is a collision step, which was transformed into a three-step operator: encoding, modulation, and decoding. It was demonstrated that different spatio-temporal patterns found in Turing two-component systems can arise in one-component networks of kinetic automata consisting of five structural modules: an encoder, a modulator, a decoder, a propagator, and storage.

https://doi.org/10.25088/ComplexSystems.25.3.195

In a subsequent paper [3], devoted primarily to the problem of morphogenesis, encoder and decoder blocks were augmented by the incorporation of four new structural elements similar to electronic analog filters. It was shown that these relatively small enhancements dramatically increase the tunability of the model and the complexity of its behavior, endowing the model with the ability to produce spatio-temporal patterns, which can be attributed to the process of morphogenesis in real biological and physical systems. The main idea behind that paper was to demonstrate that anisotropic diffusion, usually regarded as anomalous, in fact is quite ubiquitous and can be harnessed in morphogenetic engineering and robotics.

This paper is dedicated entirely to modulation, which is a key step in the kinon model, shaping most of its unusual properties. Although the CRT method, used as a computational kernel in the basic model, proved to be elegant and expressive, it is only one among many possible transformation techniques. The primary goal of this paper is to introduce a novel tensor-based method of modulation and to show that the extended modulator, augmented by a new structural element called a *kinetic tensor*, is able not only to enhance further the tunability and complexity of the model, but also to emulate many other discrete-time continuous-state dynamical systems. The comprehensive study of tensor modulation is far beyond the scope of a single paper, so it will be investigated initially from a phenomenological point of view, that is, by the identification and classification of spatio-temporal phenomena arising as a consequence of different modes or parameters of modulation in elementary (one-dimensional) kinetic automata. Special attention will be given to quantity conservation, which is the most salient feature of the model.

2. Background

The noun modulation has several meanings. It always involves some kind of deliberate modification or change. The Latin root, *modulationem*, has a musical meaning: rhythm, singing and playing, or melody. Thus, in music, modulation refers to a change in stress, pitch, loudness, or tone of the voice. In electronics and telecommunications, modulation is the variation of a property of an electromagnetic wave or signal, such as amplitude, frequency, or phase. In the kinon model [1, 3], modulation designates the transformation of a vector of input ranks $\{R\}$ into a vector of output rates $\{\mathcal{R}\}$, carried out by a respective block called a *modulator* (Figure 1).

Quantity conservation in the model is achieved due to encoder and decoder blocks, which are absent in other complex dynamical sys-

https://doi.org/10.25088/ComplexSystems.25.3.195

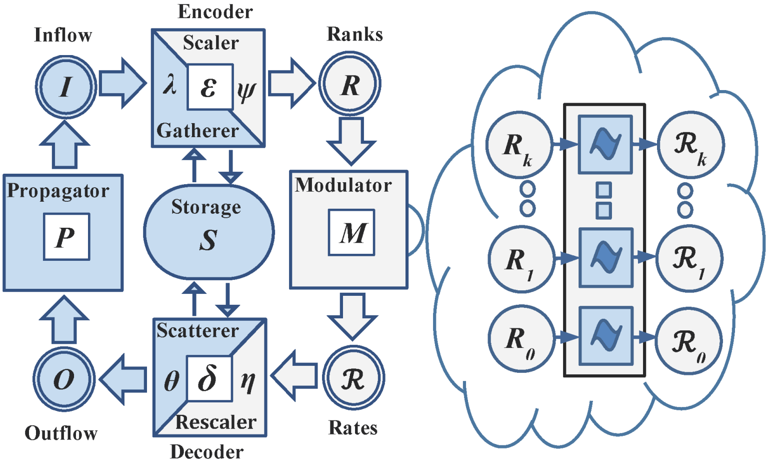


Figure 1. Kinon state-transition diagram.

tems, including the LBM. According to Rosen's modeling relation [4], they are proxies between a natural system (the neighborhood of the kinon) and a formal one (the internal kinetic model implemented by a modulator). They carry out two tasks: the conversion of absolute (raw) values into their relative (abstract) form and the conservation of the total quantity during a modulation step. Formally, kinetic transformation is a one-to-one structure-preserving mapping $R : Q^{k+1} \to Q^{k+1}$, with the following *master equation* (equation of balance):

$$\sum_{i=1}^{k} \left(I_i - O_i\right) = \Delta S, \tag{1}$$

where I_i, O_i, and S are inputs, outputs, and storage, respectively.

This equation implies that apart from topological invariance (nodes may have different valance), the model permits *stateless* kinons without storage. In this case, the right-hand side of equation (1) is always zero ($\Delta S = 0$) and all collisions are *elastic*; that is, the sums of the inputs and outputs are equal. This is analogous to Kirchhoff's circuit laws, with the difference that kinon links are bidirectional. In *stateful* kinons, collisions are *inelastic* in general ($\Delta S \neq 0$). Equation (1) is consistent with the conservation law and the kinetic theory, which was the reason for naming the model a *kinetic automaton*. Like the LBM, the model is relational and quantity conservative, because it was designed to be able to simulate real physical phenomena, for instance, fluid dynamics.

https://doi.org/10.25088/ComplexSystems.25.3.195

Quantity conservation entails additional computational costs, but the price for that is not as high as it may seem. The evolutionary steps for a one-dimensional network with a periodic boundary in continuous cellular automata (CCA) [5] and coupled map lattices (CML) [6] can be implemented in the Wolfram Language with just one line of code. The basic kinetic automaton (BKA) needs a few extra lines of code (Figure 2).

```
CCAstep[n_, a_] :=
   Map[Mod[a+#,1]&,(n+RotateRight@n+RotateLeft@n)/3];

CMLstep[n_, a_ ,ϵ_] :=
   Map[(1-a*#^2)&,(1-ϵ)n+ϵ/2(RotateRight@n+RotateLeft@n)];

BKAstep[n_,k_]:=Module[{inp,out,ri,ro,si},
(* Propagation *)
inp={n[[1]],RotateRight@n[[2]],RotateLeft@n[[3]]};
(* Encoding *)
si=Total[N@inp];                                (* Gathering *)
ri=Map[#/(si/.{0.→∞})&,inp];                    (* Scaling *)
(* Modulation *)
ro=Map[Clip[1-k*#,{0,1}]&,ri];
(* Decoding *)
out=Chop@Map[#*si/(Total[N@ro]/.{0.→∞})&,ro];   (* Rescaling *)
out[[1]]=si-Total[Rest@out];out];               (* Scattering *)
```

Figure 2. The Wolfram Language code for evolutionary steps in one-dimensional CCA, CML, and BKA.

Since quantity is preserved via storage by gathering and scattering blocks during encoding and decoding, modulation does not need to be conservative. In the CRT method, all components of the rank vector $\{R\}$ with a unit sum are mapped independently, so the output vector $\{\mathcal{R}\}$ generally does not have a unit sum. Therefore, any vector-to-vector transformation can be used for modulation. In other words, a modulator can be treated as a *black box* transforming multiple inputs into conjugate outputs.

3. Motivation

In multilinear algebra, a black box metaphor can be applied to a tensor that takes in one collection of numbers and outputs a different one. To a great extent, this paper was inspired by the concept of the *response tensor* originally presented in the context of irreversible thermodynamics by Richardson [7]. Later, it was demonstrated that all linear phenomenology of irreversible thermodynamics, dubbed *phenomenological calculus*, could be applied to a large class of complex dynamical systems and derived from three simple postulates [8]:

Postulate 1. The specification of the forces $\{\boldsymbol{F}_i\}$ acting upon a system and the set of the constitutive parameters $\left\{a^i\right\}$ conjugate to those forces is sufficient to determine a phenomenological description of the system dynamics.

Postulate 2. The system dynamics are characterized phenomenologically by the response tensor $\underline{R} = a^i \boldsymbol{F}_i$.

Postulate 3. The response tensor $\underline{R}$ is invariant under coordinate transformations in description space.

It was posited that if the dissipation function of the system is derived in the form $\delta = \boldsymbol{F}_i \,.\, \boldsymbol{J}^i$ (Einstein summation notation), then the resultant fluxes (or effects), denoted $\boldsymbol{J}^i$, are given phenomenologically as a linear function of the conjugate set of forces (or causes):

$$\boldsymbol{J}^i = L^{ij} \boldsymbol{F}_j, \tag{2}$$

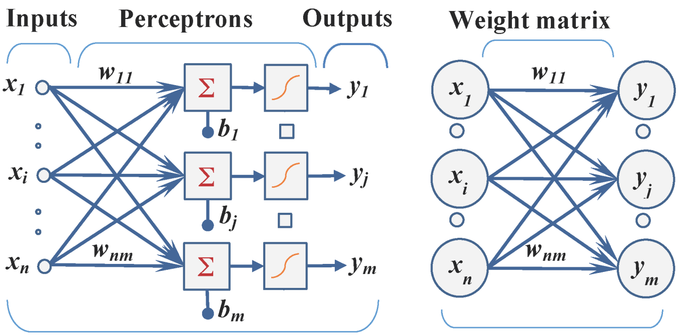
where $L^{ij} = a^i \,.\, a^j$ are the elements of a metric tensor.

In the kinon model, generalized forces $\{\boldsymbol{F}_i\}$, generated by fluxes $\left\{\boldsymbol{J}^i\right\}$ and a metric tensor L^{ij}, correspond to input ranks $\left\{\boldsymbol{R}_j\right\}$, output rates $\left\{\mathcal{R}^i\right\}$, and a modulator M^{ij}, respectively, so a modulation step can be written algebraically as follows:

$$\mathcal{R}^i = M^{ij} \boldsymbol{R}_j. \tag{3}$$

Thus, a modulator can be regarded as a metric tensor transforming *covariant* input ranks $\boldsymbol{R}_j$ into *contravariant* output rates $\mathcal{R}^i$.

A black box abstraction can also be applied to artificial neural networks (ANN) [9], which can be exemplified by a single-layer feedforward neural network consisting of *perceptrons* (Figure 3(a)). With-



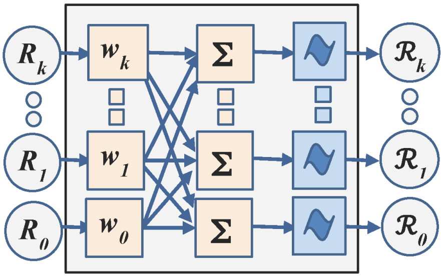
(a) $y = f(\underline{\mathbf{W}}\, x + b)$ (b) $y = \underline{\mathbf{W}}\, x$

Figure 3. Neural networks: (a) perceptrons and (b) associative memory.

out thresholding f and biasing b, it represents a *linear associative memory*, usually called a Kohonen network [10] (Figure 3(b)). It associates the output vector with the input one via a fully connected, weighted bipartite graph, which can be calculated by the inner (dot) product of the weight matrix $\underline{\mathbb{W}}$ and the input vector, so the weight matrix $\underline{\mathbb{W}}$ is fully analogous to the response tensor.

4. Tensor Modulation

The tensor approach allows capturing the behavior of a system without knowing its internal structure, so it can be applied to a wide range of complex dynamical systems, and the kinon model in particular. In order to demonstrate this, a modulator, containing only kinetic maps for each port, was augmented by a *kinetic tensor* that transforms the covariant vector of ranks into a contravariant form before transforming it via a kinetic map. Schematically, the kinetic tensor can be implemented by a set of fully interconnected splitters and couplers, which are highlighted by a peach color in Figure 4.



$$\mathcal{R} = M\,(\,\underline{\mathbb{W}}\ R\,)$$

Figure 4. Tensor modulator.

It is easy to notice the equivalence of the kinetic tensor to a linear associative memory and the close resemblance of a tensor modulator to a single-layer neural network. In accordance with their names, splitters split input values among the couplers according to the assigned weights, while couplers just couple (sum) them up. The result of this rearrangement ("entanglement") fully corresponds to the inner (dot) product of the input vector R and the weight matrix $\underline{\mathbb{W}}$. The assigned weights represent the components of the kinetic tensor. Thus, a *tensor modulator* consists of a *kinetic tensor* and a set of *kinetic maps* for each port.

https://doi.org/10.25088/ComplexSystems.25.3.195

This considerably improves the tunability of modulation, but raises the problem of dimensionality. A one-dimensional modulator has three ports (one for storage and two for neighbors), so the kinetic tensor will have nine (3×3) components. In higher dimensions, the number will grow polynomially. One of the most attractive features of the kinon model is its ability to be tuned by a single parameter of the kinetic map, which seems to be lost in tensor modulation. One of the possible solutions to the problem is parametric tensor construction, described in the following.

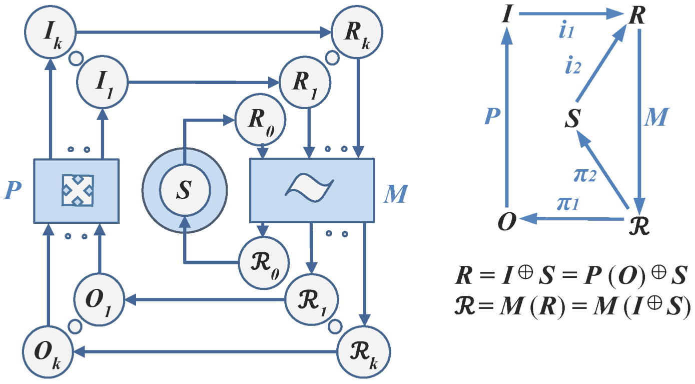
Although tensor components are independent of each other, one can dramatically reduce the parameter space by imposing interdependencies among them. This approach is used in CML, where coupling coefficients have a unit sum and are defined by a single parameter ε [6]:

$$x_{n+1}(i) = \left(1 - \varepsilon\right) f(x_n(i)) + \frac{\varepsilon}{2} \sum_{j=1}^{N} f(x_n(j)). \tag{4}$$

Similarly, one can use an arbitrary binary matrix as a template, which can be converted into a real-valued one by the substitution (fuzzification) of zeros and units with a real-valued coefficient $\varepsilon \in \left[0, 1\right]$: $\left\{0 \to \varepsilon, 1 \to \left(1 - \varepsilon\right)\right\}$. It corresponds to the coupling coefficient in CML, so can be called a *coupling rate*, but also a *fuzzy rate*, or a *fuzzifier*.

This method can be used for the construction of kinetic tensors of any rank and dimension. Low-dimensional tensors can be enumerated by integer numbers, similar to transition rules in elementary cellular automata (ECA) [11], by the flattening of tensor components into binary strings and converting them into a shorter decimal form. Three-by-three tensors can be encoded by three-digit octal numbers, in which digits represent corresponding rows of the tensor. Therefore, a kinetic tensor can be denoted as $\tau(\varepsilon)$, where τ stands for a tensor code and ε for a fuzzy rate, which can be omitted when it is equal to zero. For example, the code 421 represents the *identity* kinetic tensor with a zero fuzzy rate. A tensor modulator with the identity tensor is equivalent to a basic modulator containing only kinetic maps.

Some kinetic tensors, which can be called *conservative*, have a remarkable property that the inner (dot) product of any vector with them does not change the sum of the vector. It is important for the kinon model, because quantity conservation is its main requirement. It is rather obvious that it takes place when all vectors w_i in Figure 4 have a unit sum of components. It corresponds to the matrix W, in which all columns have a unit sum. Tensor modulation with a conservative kinetic tensor and the identity kinetic map $f(x) = x$ does not need encoding and decoding steps for quantity conservation. In this case, the basic kinon model can be reduced into a minimalistic kinon model (MKA), containing only three modules: a propagator (P), storage (S), and a tensor modulator (M) (Figure 5(a)). Using a category theory notation adopted for the description of the kinon model [3], the MKA algorithm can be represented by the categorical diagram shown in Figure 5(b). For a one-dimensional network with a periodic boundary, the evolutionary MKA step can be implemented by a single line of Wolfram Language code (Figure 5(c)).



(a) (b)

```
MKAstep[n_,t_,f_] := (* n-network, t-tensor, f-map *)
  Transpose@Map[Map[f[#]&,Dot[t,#]]&,
    Transpose@{n[[1]],RotateRight@n[[2]],RotateLeft@n[[3]]}];
```

(c)

Figure 5. MKA: (a) schematic diagram, (b) categorical diagram, (c) Wolfram Language code.

In fact, the MKA is equivalent to the LBM and can be used for its implementation, because both models are conservative. When the requirement of quantity conservation is eliminated, this model can be used for the emulation of CCA and CML as well. With the appropriate settings of the kinetic tensor $\tau(\varepsilon)$ and the kinetic map f, they can be emulated by the MKA, which will be demonstrated in the next section. While the basic kinon model is a generalization of the LBM, the MKA can be regarded as a generalization of CCA and CML. Hence, the kinon model can serve as a unifying framework for a broad range of discrete-time continuous-state dynamical systems.

5. Results

Following the long-established tradition to study complex dynamical systems with elementary (one-dimensional) models, only the simplest one-dimensional kinon network will be considered here. It consists of 101 kinons connected in a ring. The total quantity available in the network Ω is equal to 50, which is equivalent to the average value 0.5 in a homogeneous state visualized as a gray color in a grayscale image. Evolution of the network starts either from a singular configuration when only the central kinon has nonzero storage equal to Ω, or a random configuration where Ω is distributed randomly in a unit range $[0, 1]$ among the kinon storages. By default, the basic kinon model (BKA) will be used, unless another (MKA) is specified.

5.1 Classification of Behavioral Patterns

Despite different state spaces (continuous versus discrete), the behavior of elementary kinon networks is very similar to ECA; therefore, it can be classified in a similar way. The most well-known classification system of ECA dynamics includes four classes [12]:

Class I. Tends to a spatially homogeneous state.

Class II. Yields simple stable or periodic structures.

Class III. Exhibits chaotic aperiodic behavior.

Class IV. Yields complicated localized structures.

Li and Packard iteratively refined Wolfram's scheme and distinguished six classes [13, 14]. Roughly, class 1 corresponds to Wolfram class I, classes 2, 3, and 4 constitute class II, class 5 is equivalent to class III, and class 6 is class IV. Li–Packard classification is based on the differences in various statistical measures applied to asymptotic behavior, which is more convenient for the analysis of the kinon network dynamics, so it will be adopted here.

The phenomenology of pure tensor modulation, that is, with the identity kinetic map and binary kinetic tensors (ε is equal to zero), will be considered first. There are 512 binary 3×3 tensors, which are encoded by three-digit octal numbers denoting corresponding rows of the matrix with a prefix for the initial state: random (r) and singular (s). All tensors are grouped according to Li–Packard classification in Table 1 and listed in Appendix A.

Initial State	Class 1	Class 2	Class 3	Class 4	Class 5	Class 6
Random	278	188	21	21	3	1
Singular	257	192	38	14	3	8

Table 1. Li–Packard classification of 512 binary 3×3 tensors.

The most characteristic or interesting examples of each class are shown in Figures 6–11, where the upper row displays the evolution during the first 200 steps, the middle one represents the decimated images (1:101) of 10 000 evolutionary steps, and the last row, except for class 1 where all results converge to a spatially homogeneous state, shows the histograms of the final steps. Due to a periodic boundary, the decimated images, composed of the time steps multiple to the chain length (101 in our case), contain only vertical stripes or lines for any stable or globally shifted fixed patterns, which simplifies fixed-point determination.

5.1.1 Class 1: Spatially Homogeneous (Null-Point) Pattern

This class is the most ubiquitous among others and is more typical for random initial states, which is rather predictable. However, it is common for singular configurations as well, where transients can be very long or rather complicated (s066, s643) before a spatially homogeneous state is reached (Figure 6).

5.1.2 Class 2: Spatially Inhomogeneous Fixed or Globally Shifted Fixed Pattern

This class is less abundant and appears slightly more often in singular configurations (Figure 7). Singular configurations usually quickly con-

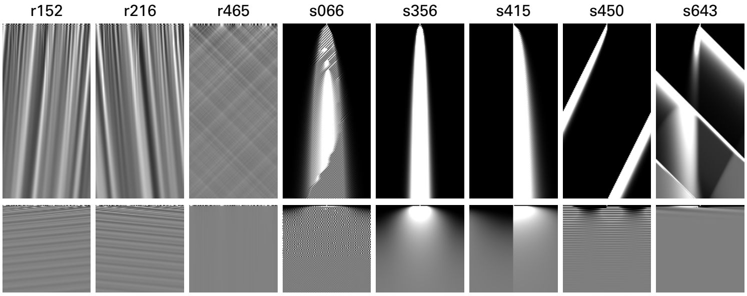


Figure 6. Class 1: spatially homogeneous (null-point) pattern.

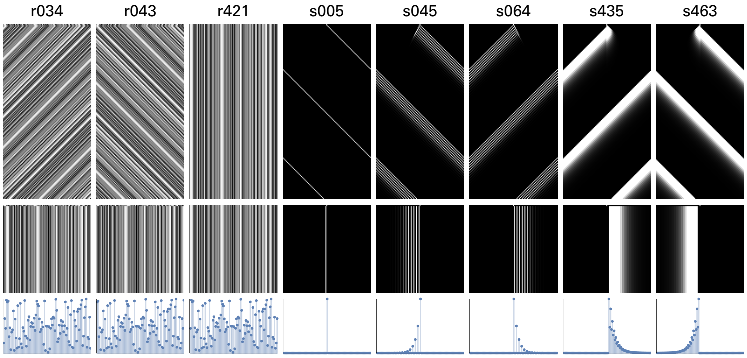


Figure 7. Class 2: spatially inhomogeneous fixed pattern.

https://doi.org/10.25088/ComplexSystems.25.3.195

verge to shifted local persistent structures similar to gliders in Conway's Game of Life: solitary spikes (s005), trains of diminishing spikes (s045, s064), or solitary asymmetric waves (s435, s463).

5.1.3 Class 3: Periodic or Globally Shifted Periodic Pattern

This class is much rarer than the previous classes. The period of these patterns consists of two or three cycles, which can be globally shifted (Figure 8). In vertical patterns, decimated images contain stripes or lines with indentations at every second or third line. In shifted patterns, decimated images consist of dashed vertical lines where gaps correspond to stationary steps. It was mentioned earlier that some tensors are quantity conservative by themselves. There are 27 conservative tensors and all of them exhibit class 2 or 3 behavior in any initial configuration. This rather boring repetitive or stationary behavior is a consequence of their strict linearity. All conservative tensors are highlighted in Appendix A.

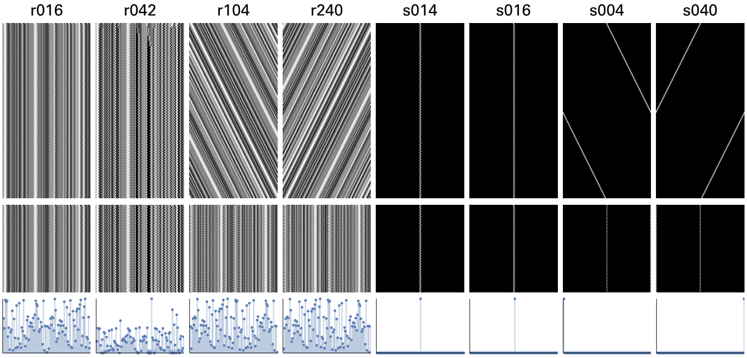


Figure 8. Class 3: periodic behavior.

5.1.4 Class 4: Periodic Pattern between Globally Shifted Walls

These patterns are quite rare, but interesting enough (Figure 9). Most of them look very similar to class 2 or 3, but closer examination reveals periodicity, often barely visible. It is manifested by the intermittency of stripes, which is seen more clearly in decimated images. Intermittency indicates two-cycle or three-cycle periodicity bounded by fixed or globally shifted walls. In rare cases, the period can vary in different domains (r014). Some singular configurations (s154, s246) deflect to the left or right from the vertical (time) axis. This phenomenon arises due to the intrinsic property of the tensor, which can be termed as its *chirality* (“handedness”), rather than to a global shift.

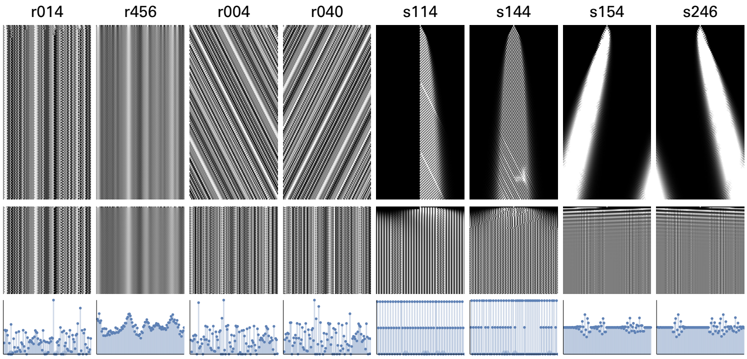


Figure 9. Class 4: periodic behavior between fixed or shifted walls.

5.1.5 Class 5: Globally Chaotic Pattern

It is the rarest of all classes and all cases are shown in Figure 10. The initial deflection due to chirality of tensors 046 and 056 eventually disappears, but the volatility remains the same, which can be visually estimated by histograms and densities of white pixels in decimated images. Tensor 044 exhibits more “uniform” chaotic behavior than tensors 046 and 054. Interestingly, they are almost the same except for their initial chirality, though they are not transposes of each other.

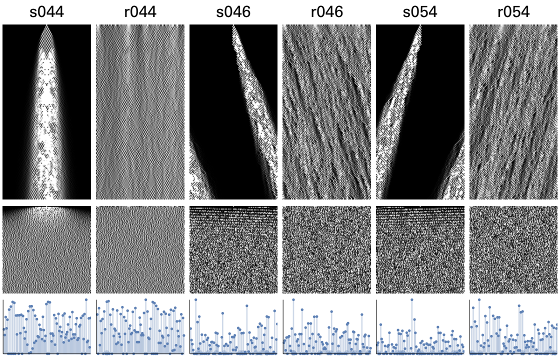


Figure 10. Class 5: chaotic behavior.

5.1.6 Class 6: Complex Pattern

This class is characterized by long transients and complex spatio-temporal patterns, including both oscillating and propagating structures (Figure 11). It includes those cases that did not fall in the previous groups because of some abnormalities. The hallmark of complexity is spatial and temporal intermittency, clearly visible in decimated images

https://doi.org/10.25088/ComplexSystems.25.3.195

and histograms (s015, s062). Configurations r065 and s465 produce “tartan” and “zigzag” patterns that look repetitive, but their periodicity is equal to the width of the image, so it is not invariant under its change. Besides, these configurations exhibit a very interesting phenomenon of elastic collisions similar to the reversible rule 122R in ECA [5, p. 442], or two-dimensional cellular automata emulating an ideal gas of particles [5, p. 447]. The configurations s542 and s614 are the most puzzling. After 100 initial immobile steps, they began to slowly but steadily deflect with a constant velocity, and after approximately 5850 steps, their movement abruptly terminates. This is possibly the artifact of real-valued computations, because with higher values of Ω, the period of movement increases and patterns become less symmetrical.

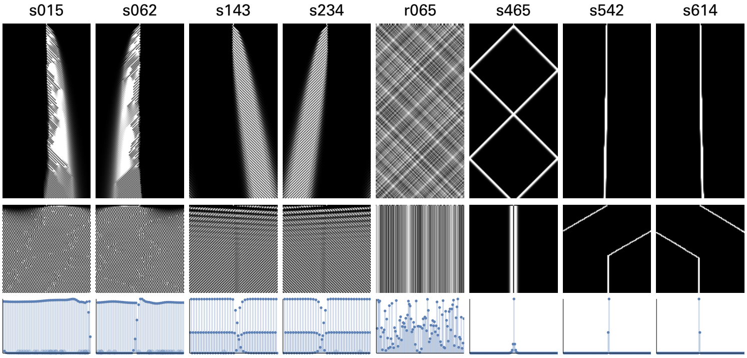


Figure 11. Class 6: complex behavior.

5.2 Phenomenology of Tensor Fuzzification

This group of results uses real-valued tensors, which are obtained through the fuzzification of binary tensor components with a real-valued parameter ε in a unit range, called a *fuzzy rate*, by the substitution $\{0 \to \varepsilon, 1 \to (1-\varepsilon)\}$. In random configurations with fuzzy rates close to zero, chaotic behavior or a disordered fixed state gradually converges to quite regular shifted stripes, which become broader with the increase of the parameter ε. After a certain critical value (usually much less than 0.1), any initial random distribution converges to a spatially homogeneous fixed point (Figure 12).

In singular configurations, tensor fuzzification always leads to homogeneity, except for marginal fuzzy rates, but can drastically affect transient states. It can change the class of behavior (Figure 13(a)) or chirality of the tensor when the fuzzy rate exceeds 0.5 (Figure 13(b)). When the fuzzy rate reaches its maximum, the tensor becomes a binary complement of the tensor template. Its dynamics often tend to non-homogeneity again, but usually, it is not the same as it was originally. They are equivalent to the dynamics of the *dual tensor*, which can be derived by the binary inversion of the tensor and replacing the fuzzy rate with the opposite value. In ECA, the reflection and Boolean conjugation symmetries are used to classify 256 transition rules into 88 equivalence classes. On the basis of the symmetry under binary inversion and transposition, all 512 binary tensors can be divided into 144 groups, which can simplify the exploration of the parameter space, although these groups are not behaviorally equivalent.

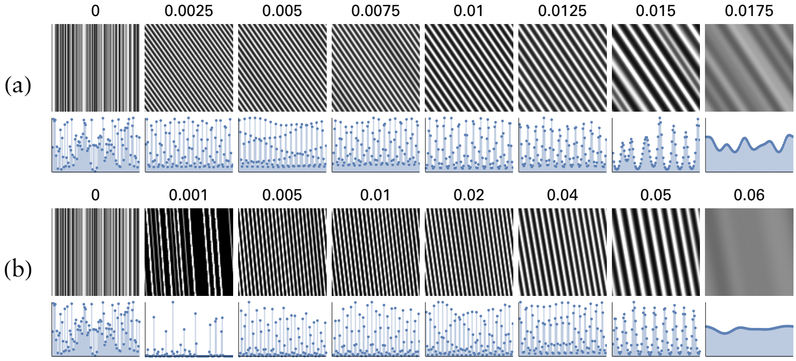


Figure 12. Phenomenology of fuzzification in (a) r002 and (b) r102.

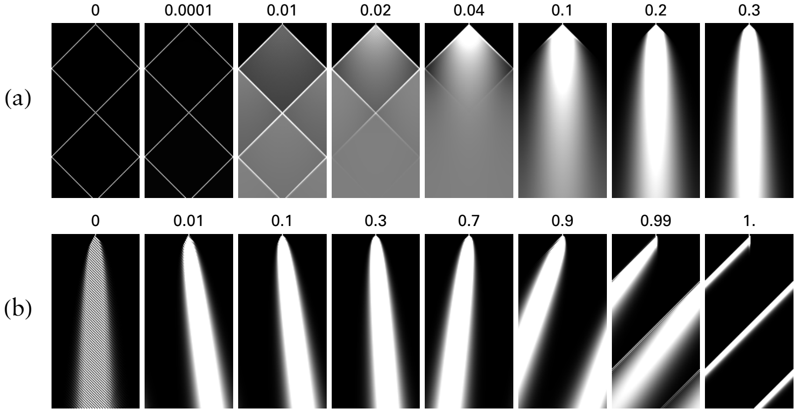


Figure 13. Phenomenology of fuzzification in s065 (a) and s343 (b).

5.3 Phenomenology of Tensor-Map Modulation

The first experiments with elementary kinetic automata [1] demonstrated that even a simple kinetic map $f(x) = \text{Max}[0, (1 - kx)]$ exhibits highly complex behavior and revealed the existence of a narrow range of the parameter k when a nearly homogeneous initial state converges to a dynamical ordered or chaotic pattern $\left(1.5 < k < 2\right)$ rather than a homogeneous $\left(k < 1.5\right)$ or stable non-homogeneous state $\left(k > 2\right)$. The most dramatic changes occur when k is near 1.6, which is confirmed by tensor modulation with the identity kinetic tensor 421 (Figure 14).

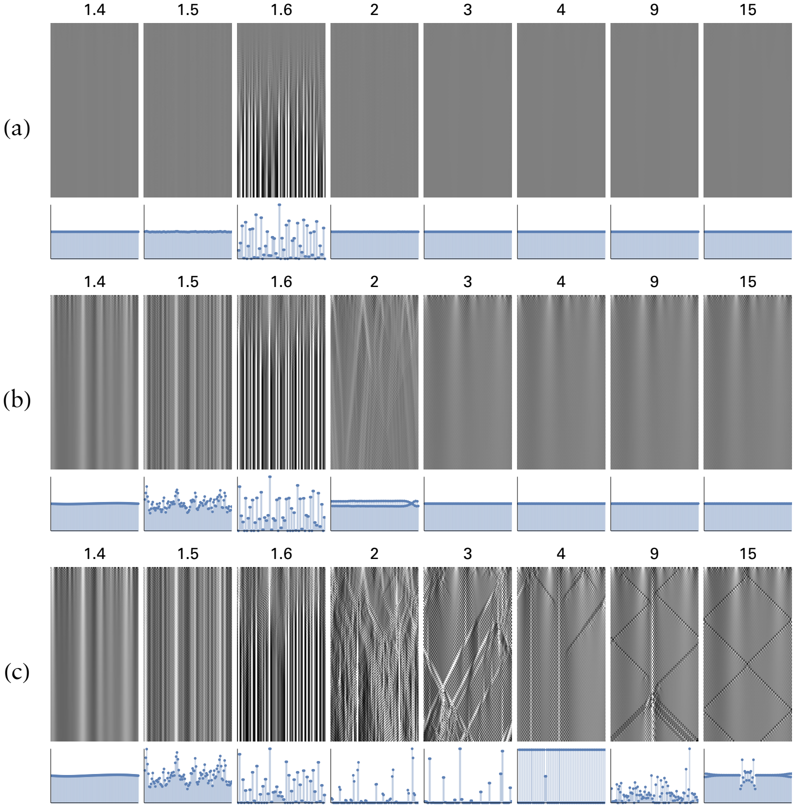


Figure 14. $f(x) = \text{Max}[0, (1 - kx)]$ modulation with different random initial states: (a) almost homogeneous [0.49, 0.51], (b) inhomogeneous [0.2, 0.8], (c) fully random [0, 1].

In singular configurations, this map behaves almost the same under small values of the parameter k. Higher values of the kinetic parameter $\left(k > 4\right)$ exhibit rather complex transient states gradually converging to chaotic patterns (Figure 15).

https://doi.org/10.25088/ComplexSystems.25.3.195

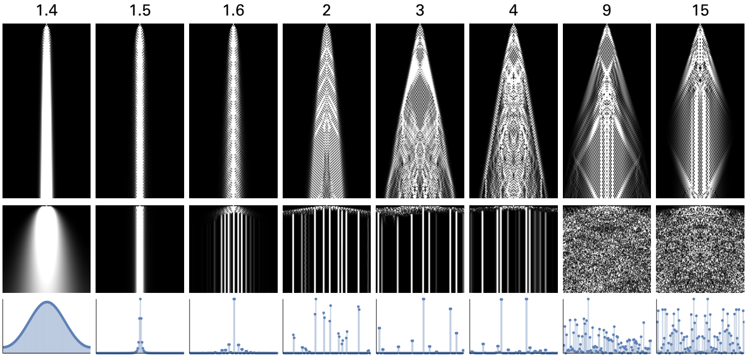


Figure 15. $f(x) = \mathrm{Max}[0, (1 - kx)]$ modulation in s421.

It is evident from the results shown that tensor and map modulations are self-sufficient to produce any class of behavior. Combined tensor-map modulation can increase further the complexity of behavior. The map $f(x) = \mathrm{Max}\left[0, \left(1 - kx\right)\right]$ confirmed its complex behavior in the narrow range $1.5 < k < 2$ with many tensors. The configuration s412 exhibits all four Wolfram classes of behavior (Figure 16): class I $\left(k < 1.5\right)$, class II $\left(k > 3.5\right)$, class III $\left(2 < k < 3.5\right)$, and class IV $\left(1.5 < k < 2\right)$. Again, the most ordered state is achieved under k equal to the “magic” number 1.6, which is strikingly close to the golden ratio $\varphi = 1.618$.

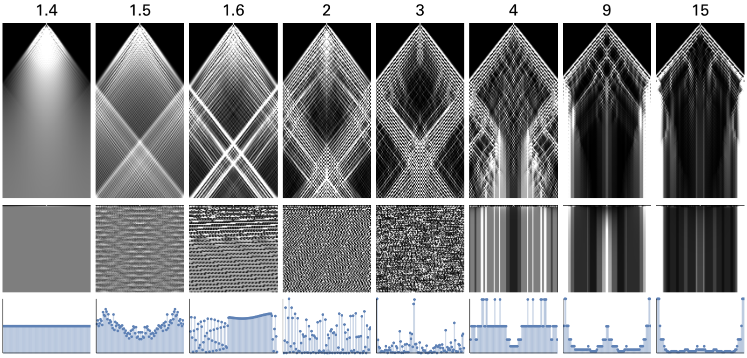


Figure 16. $f(x) = \mathrm{Max}[0, (1 - kx)]$ modulation in s412.

5.4 Phenomenology of Modulation in the Minimalistic Kinon Model

Using different kinetic tensors and maps with tunable parameters ε and k, it is possible not only to generate any behavior, but also to emulate other discrete-time continuous-state complex dynamical sys-

https://doi.org/10.25088/ComplexSystems.25.3.195

tems. For instance, the MKA with a kinetic tensor 000(1/3) and a kinetic map $f(x) = \mathrm{Mod}\left[x + k,\ 1\right]$ can emulate the behavior of CCA and generate spatio-temporal patterns, which are almost identical to those shown in [5, p. 160] (Figure 17).

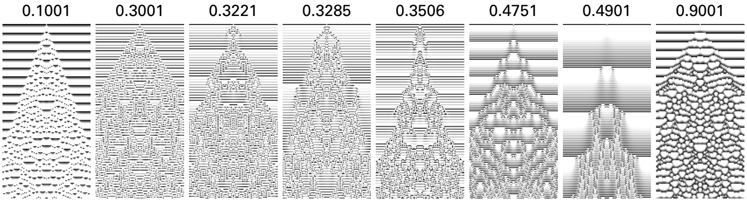


Figure 17. CCA emulation with s000(1/3) and $f(x) = \mathrm{Mod}[x + k,\ 1]$.

The canonical CML (equation (4)) with a logistic map $f(x) = 1 - a x^2$ exhibits rich phenomenology under different parameters ε and a [15]. They include frozen random patterns with spatial bifurcation and localized chaos, pattern selection with suppression of chaos, spatio-temporal intermittency, and traveling waves. Similar patterns can be obtained in the MKA with $f(x) = 1 - k x^2$ in configuration r421(ε), where zeros are substituted by $\varepsilon \big/ 2$ in order to comply with equation (4) (Figure 18). The bottom row contains decimated images (1:100) of spatio-temporal dynamics.

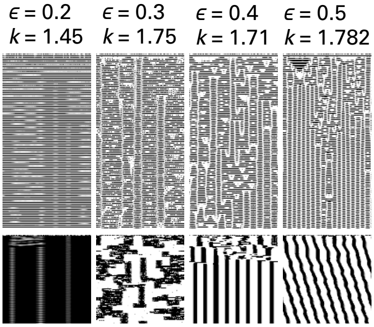

$\epsilon = 0.2$ $k = 1.45$
$\epsilon = 0.3$ $k = 1.75$
$\epsilon = 0.4$ $k = 1.71$
$\epsilon = 0.5$ $k = 1.782$

Figure 18. CML emulation by r421(ε) and $f(x) = 1 - k x^2$.

5.5 Concluding Remarks

Figures 6–18 can be reproduced in Wolfram Mathematica 9 or higher with the code provided in Appendix B. For comparability of results, all random initial configurations use the same random seed, thus are fully identical. Nevertheless, there can be slight deviations because of the peculiarities of real-valued computations on different platforms. It

https://doi.org/10.25088/ComplexSystems.25.3.195

should be also noted that changing the random seed or the range of random values can dramatically affect asymptotic behavior of tensors, which was demonstrated in Figure 14. Besides, some tensors have very long transient states (up to millions of steps), which complicates fixed-point determination. Therefore, the classification of tensors in Table 1 and Appendix A is rather tentative and might be subject to change.

6. Discussion

The results shown convincingly demonstrate the viability of a tensor approach to modulation in kinetic automata, which can enhance their future development and open new perspectives on their possible applications.

Tensor calculus was developed around 1890 by Ricci-Curbastro and Levi-Civita under the name of absolute differential calculus. In the twentieth century, the subject came to be known as tensor analysis and achieved broader acceptance with the introduction of Einstein's theory of general relativity in 1916. Electrical engineer Gabriel Kron pioneered the application of tensors to the analysis of electrical networks [16]. Tensors were found to be useful in continuum mechanics, where stress tensor, strain tensor, and elasticity tensor are widely used. In 1980, Pellionisz and Llinás introduced the tensor network theory to describe the behavior of the cerebellum in transforming afferent sensory inputs into efferent motor outputs. They proposed that intrinsic central nervous system space could be described and modeled by an extrinsic network of tensors that together describe the behavior of the central nervous system. By treating the brain as a "geometrical object," brain function could be quantified and described as a network of tensors [17, 18].

Around the same period in the 1980s, tensor calculus was applied to the phenomenology of complex dynamical systems under the name of phenomenological calculus, which was described earlier. Initially, it considered only linear systems that implicitly assume that cause and effect are always collinear and each cell is homogeneous and isotropic with respect to diffusion in all directions. In 2006, phenomenological calculus was extended by Louie and Richardson into a much broader domain of *anisotropy* [19]. In their own words: "It may even be argued that nature is anisotropic. Isotropic systems are simply 'weakly anisotropic' ones, for which linear approximations suffice." Essentially, anisotropic systems are what the kinon model was intended for. Anisotropy, as opposed to isotropy, implies that the flux is not always aligned with the force. In other words, they are directionally independent. Anisotropic systems have much more complex cause-effect phenomenology than isotropic ones and can be found wherever a wave propagates through a medium: in constrained transport, crystallography, fluid dynamics, seismology, cosmology, and others.

In anisotropic systems, the fluxes cannot be derived from the forces by simple scalar multiplication and must be "independently scaled." That is why a modulator in the basic kinon model simply transforms input ranks into output rates via a nonlinear kinetic map on a one-by-one basis. In phenomenological calculus, the components of the response tensor are scalar products of forces and constitutive parameters of the system; hence the response tensor is a *dyadic* (second-order) tensor. Louie and Richardson argue that a *triadic* (third-order) response tensor is needed for anisotropy; that is, the constitutive parameters of the anisotropic system must be tensors themselves. The current implementation of tensor modulation employs only dyadic kinetic tensors. The desired anisotropy of modulation is achieved via the final nonlinear map modulation.

The idea of representing the internal structure of a complex multiport system by a matrix and calculating its output by a simple vector-matrix multiplication is rather old and appeared in many contexts. It can be traced back a formulation of quantum mechanics called matrix mechanics, created by Heisenberg, Born, and Jordan in 1925. Later, it was transformed into S-matrix theory, which was very influential in the 1960s but now is largely abandoned. However, under the name of the S-parameter approach, it is still alive and thriving in microwave engineering, where an electrical network is regarded as a "black box" that interacts with other circuits through ports [20]. The network is characterized by a square matrix of complex numbers called an S-parameter matrix, which is used to calculate its response to signals applied to the input ports.

It was already mentioned that a black box abstraction can also be attributed to ANN, which are based on the discrete-time logic-based neural model invented by McCulloch and Pitts in 1943 [21]. However, one of the first neuron models was proposed by Rashevsky a decade earlier [22]. His idea was to use a pair of linear differential equations and a nonlinear threshold operator to account for peripheral nerve excitation. Rosen showed [23, 24] that Rashevsky's two-factor systems are closely related to Turing's theory of morphogenesis [25] but are capable of far wider application. Despite many advantages of discrete models, the continuous Rashevsky's neuron model might be suitable for modeling whole masses of neurons. Though a tensor modulator looks much like a single-layer feed-forward neural network, the whole kinon would be better viewed as a model of the arbitrary ensemble of neurons. In this sense, it is closer to Rashevsky's neuron model and Pellionisz–Llinás tensor networks, but its applicability is much broader than neural modeling.

It was shown that the MKA with a tensor modulator is able to emulate CCA, which are closely related to cellular neural networks (CNN) introduced by Chua and Yang in 1988 [26, 27]. Basically, CNN and ANN are the same, with the difference that interactions are allowed only between neighboring cells, similar to CCA. Therefore, CNN can be readily emulated by kinetic automata.

Schematically, CNN is an array of locally interconnected simple analog circuits consisting of a linear capacitor, a nonlinear voltage-controlled current source, and a few resistive linear circuit elements. The local interconnection enables efficient very large-scale integration implementations (VLSI), and many CNN chips have been reported since the first CNN chip was presented in 1991. In 1993, the CNN architecture was augmented by programmable weights, local storage, and logic, and thus became analogic (analog and logic) [28]. It was called a CNN universal machine (CNN-UM), as it has been proven that it is as universal as a Turing machine. Among the numerous applications of CNNs, image processing is the most widespread, though recent studies have proved that they can be also used for simulations in fluid dynamics and statistical physics. Since kinetic automata are analog (continuous state) dynamical systems with a comparable architecture, CNN-UM can be used for their efficient VLSI implementation.

Apart from Wolfram and Li–Packard classifications of ECA transition rules considered earlier, Adamatzky and Wuensche proposed recently to classify them by mapping onto the cognitive control versus schizotypy spectrum phase space and interpreting cellular automaton behavior in terms of creativity [29]. They found that null- and fixed-point ECA rules lie in the "autistic" domain and chaotic rules are "schizophrenic." It is rather discouraging that there are no highly creative ECA rules, and rules closest to creativity domains are two-cycle rules exhibiting wave-like patterns. At the current stage of the kinon model exploration, it is impossible to make any definite conclusion about its creativity. Nevertheless, due to the flexibility of the kinon model and the infinity of its parameter space, it can be hoped that truly "creative" configurations and parameters, capable of generating "artistic" patterns of nature, might be found in the future.

7. Conclusion

The main goal of this paper is to demonstrate the applicability of the tensor approach to modulation in kinetic automata. It was shown that the proposed tensor-based modulator is capable of generating a variety of spatio-temporal patterns, which can be attributed to all classes of behavior found in elementary cellular automata and coupled map lattices (CML): chaos, periodicity, period doubling, intermittency, domains and walls, soliton propagation, phase shifts, and others. Some of them can be qualified as complex and need further investigation.

The proposed method of tensor construction from binary templates alleviates the problem of dimensionality of the parameter space and is valid for tensors of any rank and dimension. The binary structure of tensor templates enables their enumeration and shorthand denotation in a way that is convenient both for referencing and comparison. Parametric fuzzification corresponds to the fine-tuning of the kinetic tensor. Combined tensor and map modulation can produce any dynamics with just a few parameters.

Tensor modulation not only increases the complexity of the model behavior but also permits bypassing encoding and decoding steps in some cases. It was shown that the minimalistic kinon model, consisting of only three modules: a tensor modulator, a propagator, and storage, is sufficient to emulate the behavior of continuous cellular automata and CML. Cellular neural networks can be readily emulated by kinetic automata as well. It follows that the kinon model can serve as a unifying framework for a broad range of discrete-time continuous-state complex dynamical systems.

Acknowledgments

I would like to thank A. H. Louie for his advice and helpful discussion during the writing of the paper and I. W. Richardson for his insightful concept of the response tensor.

Appendix

A. Classes of Tensor Behavior

These are the lists of binary tensors grouped in six classes of asymptotic behavior according to Li–Packard classification. In classes 2–4, tensors with globally shifted patterns are underlined, while conservative tensors are highlighted by the orange color.

Class 1. Spatially homogeneous (null) fixed patterns

r: 015,017,036,037,053,055,057,062,066,072,073,076,077,105,107,114,115,116,
117,125,127,135,136,137,143,144,145,146,147,152,153,154,155,156,157,162,163,
165,166,167,172,173,175,176,177,215,216,217,234,235,236,237,242,244,246,252,
253,254,255,256,257,260,261,262,263,264,265,266,267,270,271,272,273,274,275,
276,277,305,307,314,315,316,317,325,327,334,335,336,337,342,343,344,345,346,

347,352,353,354,355,356,357,360,361,362,363,364,365,366,367,370,371,372,373,
374,375,376,377,404,406,415,417,424,426,436,437,440,441,444,446,450,451,453,
454,455,457,462,465,466,472,473,476,477,504,505,506,507,514,515,516,517,524,
525,526,527,534,535,536,537,540,541,542,543,544,545,546,547,550,551,552,553,
554,555,556,557,562,563,564,565,566,567,572,573,574,575,576,577,604,606,614,
615,616,617,624,626,634,635,636,637,640,641,642,643,644,645,646,647,650,651,
652,653,654,655,656,657,660,661,662,663,664,665,666,667,670,671,672,673,674,
675,676,677,704,705,706,707,714,715,716,717,724,725,726,727,734,735,736,737,
740,741,742,743,744,745,746,747,750,751,752,753,754,755,756,757,760,761,762,
763,764,765,766,767,770,771,772,773,774,775,776,777
s: 017,036,037,053,055,057,066,072,073,076,077,105,107,115,116,117,125,127,
135,136,137,145,146,147,153,155,156,157,162,163,165,166,167,172,173,175,176,
177,215,217,235,236,237,252,253,254,255,256,257,260,261,262,263,264,265,266,
267,270,271,272,273,274,275,276,277,305,307,315,316,317,325,327,335,336,337,
344,345,346,347,352,353,354,355,356,357,360,361,362,363,364,365,366,367,370,
371,372,373,374,375,376,377,404,406,415,417,424,426,436,437,440,441,444,446,
450,451,453,454,455,457,462,466,472,473,476,477,504,505,506,507,514,515,516,
517,524,525,526,527,534,535,536,537,540,541,543,544,545,546,547,550,551,553,
554,555,556,557,562,563,564,565,566,567,572,573,574,575,576,577,604,606,615,
617,624,626,634,635,636,637,640,641,642,643,644,645,646,647,650,651,652,653,
654,655,656,657,660,661,662,663,664,665,666,667,670,671,672,673,674,675,676,
677,704,705,706,707,714,715,716,717,724,725,726,727,734,735,736,737,740,741,
742,743,744,745,746,747,750,751,752,753,754,755,756,757,760,761,762,763,764,
765,766,767,770,771,772,773,774,775,776,777

Class 2. Inhomogeneous fixed or shifted fixed patterns

r: 000,001,002,003,005,007,010,011,012,013,020,021,022,023,025,027,030,031,
032,033,034,035,043,045,047,060,061,063,064,067,070,071,074,075,100,101,102,
103,110,111,112,113,120,121,122,123,130,131,132,133,134,160,161,164,170,171,
174,200,201,202,203,205,207,210,211,212,213,220,221,222,223,225,227,230,231,
232,233,243,245,247,300,301,302,303,310,311,312,313,320,321,322,323,330,331,
332,333,400,401,402,403,405,407,410,411,412,413,414,420,421,422,423,425,427,
430,431,432,433,434,435,442,443,445,447,460,461,463,464,467,470,471,474,475,
500,501,502,503,510,511,512,513,520,521,522,523,530,531,532,533,560,561,570,
571,600,601,602,603,605,607,610,611,612,613,620,621,622,623,625,627,630,631,
632,633,700,701,702,703,710,711,712,713,720,721,722,723,730,731,732,733
s: 000,001,002,003,005,007,010,011,012,013,020,021,022,023,025,027,030,031,
032,033,034,035,043,045,047,060,061,063,064,067,070,071,074,075,100,101,102,
103,110,111,112,113,120,121,122,123,130,131,132,133,134,152,160,161,164,170,
171,174,200,201,202,203,205,207,210,211,212,213,216,220,221,222,223,225,227,
230,231,232,233,243,245,247,300,301,302,303,310,311,312,313,320,321,322,323,
330,331,332,333,400,401,402,403,405,407,410,411,412,413,414,420,421,422,423,
425,427,430,431,432,433,434,435,442,443,445,447,460,461,463,464,467,470,471,
474,475,500,501,502,503,510,511,512,513,520,521,522,523,530,531,532,533,552,
560,561,570,571,600,601,602,603,605,607,610,611,612,613,616,620,621,622,623,
625,627,630,631,632,633,700,701,702,703,710,711,712,713,720,721,722,723,730,
731,732,733

Class 3. Periodic or shifted periodic patterns

r: 016,042,052,104,106,124,126,142,214,240,241,250,251,304,306,324,326,340,
341,350,351
s: 004,006,014,016,024,026,040,041,042,050,051,052,104,106,124,126,140,141,
142,150,151,204,206,214,224,226,240,241,250,251,304,306,324,326,340,341,350,
351

https://doi.org/10.25088/ComplexSystems.25.3.195

Class 4. Periodic patterns between fixed or shifted walls

r: 004,006,014,024,026,040,041,050,051,056,140,141,150,151,204,206,224,226,
416,452,456

s: 056,114,144,154,242,244,246,314,334,342,343,416,452,456

Class 5. Chaotic patterns

r: 044,046,054

s: 044,046,054

Class 6. Complex patterns

r: 065

s: 015,062,065,143,234,465,542,614

B. Wolfram Language Code for Section 5

```
(* evolutionary step: n-network,
t-kinetic tensor,  f-kinetic map,
   w-weighting (T/F), q-quantity conservation (T/F) *)
step[n_, t_, f_, w_, q_] := Module[{inp, out, si, s},
    inp = {n[[1]], RotateLeft@n[[2]], RotateRight@n[[3]]};
    out = Transpose@Map[Dot[t, #] &, Transpose@inp];
    If[Head[f] == Function,
     out = Map[f[If[w, #/(Total[N@out] /. {0. → ∞}), #]] &, out]];
    If[q, si = Total[inp];
     s = si/(Total[N@out] /. {0. → ∞});
     out = Chop@Map[# s &, out];
     out[[1]] = si - Total[Rest@out]];
    out];

(* initial state: s-singular,
r-random, u-uniform, n-non-uniform *)
init[s_] := If[s == "s",
    Table[If[i == 1 && j == 51, 50, 0.], {i, 3}, {j, 101}],
  SeedRandom[123]; Prepend[Table[0, {2}, {101}],
  RandomReal[Switch[s, "r",
       {0, 1}, "u", {.49, .51}, _, {.2, .8}], 101]]];

(* tensor construction: τ-tensor code,  ε-fuzzy rate *)
tensor[τ_, ε_] :=
   Partition[IntegerDigits[FromDigits[τ, 8], 2, 9], 3] /.
  If[ε < 0, {1 → (1 - Abs[ε]), 0 → Abs[ε]/2}, {1 → (1 - ε), 0 → ε}];

(* evolution: p-parameter list *)
evolve[p_] := Module[{c, t, f, w, q},
  c = init@StringTake[p[[1]], 1]; (* initial state *)
  t = tensor[StringTake[p[[1]], -3], p[[2]]]; (* tensor *)
  f = If[Length@p > 2, p[[3]] /. {k → p[[4]]}]; (* map *)
```

```
 w = If[Length@p < 5, True, p[[5]]]; (* weighting: T/F *)
 q = If[Length[p] < 6, True, p[[6]]];
   (* quantity conservation: T/F *)
 NestList[step[#, t, f, w, q] &, c, 10 000]];

(* image generation: d-data, r-desimation rate *)
image[d_, r_] :=
  Image[Take[Total[d, {2}], {1, -1, r}], ImageSize → 101];

(* histogram plotting: s-state data *)
plot[s_] := Module[{l = Total@s},

   ListLinePlot[l, Filling → Axis, ImageSize → 101, Ticks → None,
 PlotRange → {{0, 101}, {Min[l, 0], Max[l, 1]}}]];

(* figure plotting *)
figure[p_] := Module[{d}, Grid@Transpose@Map[(
 d = evolve[Rest@#]; (* dynamics data *)
 {If[ListQ@#[[1]],
           Column@#[[1]], #[[1]]], (* caption *)
 image[Take[d, 200], 1], (* first 200 steps *)
 image[Take[d, -100], 1], (* last 100 steps *)
 image[d, If[Length@#[[1]] < 3, 101, #[[1, 3]]]],
          (* synopsis *)
 plot@d[[-1]]}) &, p]]; (* last step histogram *)

(* figures in Section 5 *)
fig6 :=  figure[Map[{#, #, 0} &,
 {"r152", "r216", "r465", "s066", "s356", "s415",
     "s450", "s643"}]];
fig7 := figure[Map[{#, #, 0} &,
 {"r034", "r043", "r421", "s005", "s045", "s064",
     "s435", "s463"}]];
fig8 := figure[Map[{#, #, 0} &,
 {"r016", "r042", "r104", "r240", "s014", "s016",
     "s004", "s040"}]];
fig9 := figure[Map[{#, #, 0} &,
 {"r004", "r040", "r416", "r456", "s114", "s144",
     "s154", "s246"}]];
fig10 := figure[Map[{#, #, 0} &,
 {"s044", "r044", "s046", "r046", "s054", "r054"}]];
fig11 := figure[Map[{#, #, 0} &,
 {"s015", "s062", "s143", "s234", "r065", "s465",
     "s542", "s614"}]];
fig12a := figure[Map[{#, "r002", #} &,
 {0, 0.0025, 0.005, 0.0075, 0.01, 0.0125, 0.015, 0.0175}]];
fig12b := figure[Map[{#, "r102", #} &,
 {0, 0.001, 0.005, 0.01, 0.02, 0.04, 0.05, 0.06}]];
fig13a := figure[Map[{#, "s065", #} &,
 {0, 0.0001, 0.01, 0.02, 0.04, 0.1, 0.2, 0.3}]];
fig13b := figure[Map[{#, "s343", #} &,
 {0, 0.01, 0.1, 0.3, 0.7, 0.9, 0.99, 1.0}]];
```

```
fig14a := figure[Map[
    {#, "u421", 0, Clip[1 - k #, {0, 1}] &, #} &,
  {1.4, 1.5, 1.6, 2, 3, 4, 9, 15}]];
fig14b := figure[Map[
    {#, "n421", 0, Clip[1 - k #, {0, 1}] &, #} &,
  {1.4, 1.5, 1.6, 2, 3, 4, 9, 15}]];
fig14c := figure[Map[
    {#, "r421", 0, Clip[1 - k #, {0, 1}] &, #} &,
  {1.4, 1.5, 1.6, 2, 3, 4, 9, 15}]];
fig15 := figure[Map[
    {#, "s421", 0, Clip[1 - k #, {0, 1}] &, #} &,
  {1.4, 1.5, 1.6, 2, 3, 4, 9, 15}]];
fig16 := figure[Map[
    {#, "s412", 0, Clip[1 - k #, {0, 1}] &, #} &,
  {1.4, 1.5, 1.6, 2, 3, 4, 9, 15}]];
fig17 := figure[Map[
    {#, "s000", 1/3., Mod[k + #, 1] &, #, False, False} &,
  {.1001, .3001, .3221, .3285, .3506, .4751, .4901, .9001}]];
fig18 := figure[Map[
    {#, "r421", -#[[1]], 1 - k #^2 &, #[[2]], False, False} &,
  {{0.2, 1.45, 100}, {.3, 1.75, 100},
     {.4, 1.71, 100}, {.5, 1.782, 100}}]];
```

References

[1] Y. Shalygo, "The Kinetic Basis of Self-Organized Pattern Formation," in *Proceedings of the 14th International Conference on the Synthesis and Simulation of Living Systems (ALife 14)*, (H. Sayama, J. Rieffel, S. Risi, R. Doursat, and H. Lipson, eds.), Cambridge, MA: MIT Press, 2014 pp. 665–672. doi:10.7551/978-0-262-32621-6-ch106.

[2] B. Chopard, P. Luthi, and A. Masselot, "Cellular Automata and Lattice Boltzmann Techniques: An Approach to Model and Simulate Complex Systems," *Advances in Complex Systems*, **5**(2), 2002 pp. 103–246. doi:10.1142/S0219525902000602.

[3] Y. Shalygo, "The Kinetic Basis of Morphogenesis," in *Proceedings of the 13th European Conference on Artificial Life (ECAL 2015)*, York, UK, Cambridge, MA: MIT Press, 2015 pp. 122–129. doi:10.7551/978-0-262-33027-5-ch027.

[4] R. Rosen, *Life Itself: A Comprehensive Inquiry into the Nature, Origin, and Fabrication of Life,* New York: Columbia University Press, 1991.

[5] S. Wolfram, *A New Kind of Science,* Champaign, IL: Wolfram Media, Inc., 2002.

[6] K. Kaneko, "Pattern Dynamics in Spatiotemporal Chaos: Pattern Selection, Diffusion of Defect and Pattern Competition Intermettency," *Physica D: Nonlinear Phenomena*, **34**(1–2), 1989 pp. 1–41. doi:10.1016/0167-2789(89)90227-3.

[7] I. W. Richardson, "The Metrical Structure of Aging (Dissipative) Systems," *Journal of Theoretical Biology*, **85**(4), 1980 pp. 745–746. doi:10.1016/0022-5193(80)90269-6.

[8] I. W. Richardson, A. H. Louie, and S. Swaminathan, "A Phenomenological Calculus for Complex Systems," *Journal of Theoretical Biology*, **94**(1), 1982 pp. 61–76. doi:10.1016/0022-5193(82)90329-0.

[9] R. Rojas, *Neural Networks: A Systematic Introduction*, Berlin, New York: Springer-Verlag, 1996.

[10] T. Kohonen, *Associative Memory: A System-Theoretical Approach,* New York: Springer-Verlag, 1977.

[11] S. Wolfram, "Statistical Mechanics of Cellular Automata," *Reviews of Modern Physics*, **55**(3), 1983 pp. 601–644. doi:10.1103/RevModPhys.55.601.

[12] S. Wolfram, "Universality and Complexity in Cellular Automata," *Physica D: Nonlinear Phenomena,* **10**(1–2), 1984 pp. 1–35. doi:10.1016/0167-2789(84)90245-8.

[13] W. Li and N. Packard, "The Structure of the Elementary Cellular Automata Rule Space," *Complex Systems*, **4**(3), 1990 pp. 281–297. http://www.complex-systems.com/pdf/04-3-3.pdf.

[14] W. Li, N. H. Packard, and C. G. Langton, "Transition Phenomena in Cellular Automata Rule Space," *Physica D: Nonlinear Phenomena*, **45**(1–3), 1990 pp. 77–94. doi:10.1016/0167-2789(90)90175-O.

[15] K. Kaneko and T. Yanagita, "Coupled Maps," *Scholarpedia*, **9**(5), 2014 4085. doi:10.4249/scholarpedia.4085.

[16] G. Kron, *Tensor Analysis of Networks*, New York: J. Wiley & Sons, Inc., 1939.

[17] A. Pellionisz and R. Llinás, "Tensorial Approach to the Geometry of Brain Function: Cerebellar Coordination via a Metric Tensor," *Neuroscience*, **5**(7), 1980 pp. 1125–1138. doi:10.1016/0306-4522(80)90191-8.

[18] A. Pellionisz and R. Llinás, "Tensor Network Theory of the Metaorganization of Functional Geometries in the Central Nervous System," *Neuroscience*, **16**(2), 1985 pp. 245–273. doi:10.1016/0306-4522(85)90001-6.

[19] A. H. Louie and I. W. Richardson, "A Phenomenological Calculus for Anisotropic Systems," *Axiomathes*, **16**(1), 2006 pp. 215–243. doi:10.1007/s10516-005-4697-5.

[20] K. Kurokawa, "Power Waves and the Scattering Matrix," *IEEE Transactions on Microwave Theory and Techniques*, **13**(2), 1965 pp. 194–202. doi:10.1109/TMTT.1965.1125964.

[21] W. S. McCulloch and W. H. Pitts, "A Logical Calculus of the Ideas Immanent in Nervous Activity," *Bulletin of Mathematical Biophysics*, **5**(4), 1943 pp. 115–133. doi:10.1007/BF02478259.

[22] N. Rashevsky, "Outline of a Physico-Mathematical Theory of Excitation and Inhibition," *Protoplasma*, **20**(1), 1933 pp. 42–56. doi:10.1007/BF02674811.

[23] R. Rosen, "Two-Factor Models, Neural Nets, and Biochemical Automata," *Journal of Theoretical Biology*, **15**(3), 1967 pp. 282–297. doi:10.1016/0022-5193(67)90138-5.

[24] R. Rosen, "Turing's Morphogens, Two-Factor Systems and Active Transport," *The Bulletin of Mathematical Biophysics*, **30**(3), 1968 pp. 493–499. doi:10.1007/BF02476609.

[25] A. M. Turing, "The Chemical Basis of Morphogenesis," *Philosophical Transactions of the Royal Society B*, **237**(641), 1952 pp. 37–72. doi:10.1098/rstb.1952.0012.

[26] L. O. Chua and L. Yang, "Cellular Neural Networks: Theory," *IEEE Transactions on Circuits and Systems*, **35**(10), 1988 pp. 1257–1272. doi:10.1109/31.7600.

[27] L. O. Chua and L. Yang, "Cellular Neural Networks: Applications," *IEEE Transactions on Circuits and Systems*, **35**(10), 1988 pp. 1273–1290. doi:10.1109/31.7601.

[28] T. Roska and L. O. Chua, "The CNN Universal Machine: An Analogic Array Computer," *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, **40**(3), 1993 pp. 163–173. doi:10.1109/82.222815.

[29] A. Adamatzky and A.Wuensche, "On Creativity and Elementary Cellular Automata," *Complex Systems*, **22**(4), 2013 pp. 361–375. http://www.complex-systems.com/pdf/22-4-2.pdf.