

# Review of Cellular Automata Approaches to Biological Modeling and Pattern Adaptation

Yogesh Phalak\* & Jonathan Zelaya\*

December 11, 2023

## Abstract

The paper chosen to study for the final project for this course is ‘Cellular Automat Approaches to Biological Modeling’ [1] by G. Bard Ermentrout and Leah Edelstein-Keshet. This was published in the Journal of Theoretical Biology in 1993. This work explores a number of biologically motivated cellular automata (CA) that arise in models of oscillatory and excitable media, developmental biology, neurobiology, and population biology. The author suggests theoretical arguments that allow for greater speed and realism in computation, as well as apply these to examples of pattern formation. We extend the authors’ work by including the tool of pattern identification using convolution in image processing to understand the mechanism of pattern adaptation.

## 1 Introduction

The work to understand the natural world has led to the development of complex mathematical models. Realistic models based on physical laws generally result in large systems of non-linear partial differential equations, such as biological phenomena. Although computer simulations offer the capability to understand these processes, these models lead to formidable numerical problems that require huge memories on fast computers. This means that it becomes very difficult and time-consuming to parametrically explore the system. One reduced-order modeling technique for these types of systems is to mimic the physical laws by a series of simple rules that are significantly easier to compute. These simple systems will be called cellular automata (CA). A cellular automaton is a collection of cells arranged in a lattice of specified shape, such that each cell changes state as a function of time, according to a defined set of rules driven by the states of neighboring cells. CA consists of simulations that are discrete in time, space, and state.

The work of CA to describe biological phenomena has been underwhelming to the point of this paper, although the spatial and temporal patterns from biology are diverse and fascinating. The author of this paper complements analytical partial differential equations (PDEs) with a number of biologically motivated automata in order to support the claim that CA is an adequate model to generalize and capture a large amount of complicated biological phenomena. In this case, using CA to examine spatial and temporal pattern formation<sup>1</sup>.

In this work, only deterministic automata are investigated. In deterministic automata, the spatial domain of the model is divided into a fixed lattice where each lattice point has a state associated with it. The state of the lattices at the next time step is determined solely by the earlier states of the cell and its neighbors. This type of automation closely resembles an evolution equation such as a PDE or integral equation. This type of CA is applicable in modeling waves in oscillatory and excitable media, predator-prey models, and spatial pattern formation.

## 2 Deterministic Automata

A majority of CAs represent an abstraction of PDEs that aim to model the spatial and temporal patterns of the given biological phenomena. Thus fall under the classification of deterministic CA. All calculations and simulations were done using Python.

---

\*Department of Biomedical Engineering and Mechanics, Virginia Tech

<sup>1</sup>The source code is available at: <https://github.com/YogeshPhalak/Natural-Pattern-Adaptation-CA>

## 2.1 General model

Active media mimic the behavior of nerve cells, muscle cells, cardiac function, and chemical reactions. This type of medium is one in which there is a unique stable rest state where small perturbations relax to this state. Large perturbations lead to an excited state before relaxing to a rest state (recovery). For a certain amount of time during the recovery period, the medium is unable to be re-excited regardless of the size of the perturbation (refractory period). Assuming each cell in a fixed array is an active CA, a simple excitable rule has three states 0,  $E$ , and  $R$ , representing the resting state, the excited state, and the refractory state, respectively. If a cell is in state  $E$  then at the next time step it becomes  $R$  and after that, it becomes 0. If a cell is in state 0 and at least one neighbor is in state  $E$  then at the next time step, the cell is put in in-state  $E$ . The rules can be represented as follows,

$$\begin{aligned} E &\rightarrow \Delta t \rightarrow R \\ R &\rightarrow \Delta t \rightarrow 0 \\ 0 &\rightarrow \Delta t \rightarrow E \text{ if } N_E \geq 1 \end{aligned} \tag{1}$$

where  $N_E$  is the the number of neighboring cells in state  $E$ , and  $\Delta t$  is the time step size.

All modifications of active CA are based on this simple rule. A typical pattern is a square-rotating spiral wave. This active CA has six states, 0, 1, 2, 3, 4, 5. The rules are,

$$\begin{aligned} 1 &\rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 0 \\ 0 &\rightarrow 1 \text{ if } N_1 \geq 1. \end{aligned} \tag{2}$$

The author's results and our own for a spiral wave can be seen in Fig.1 which gives the parameters used.

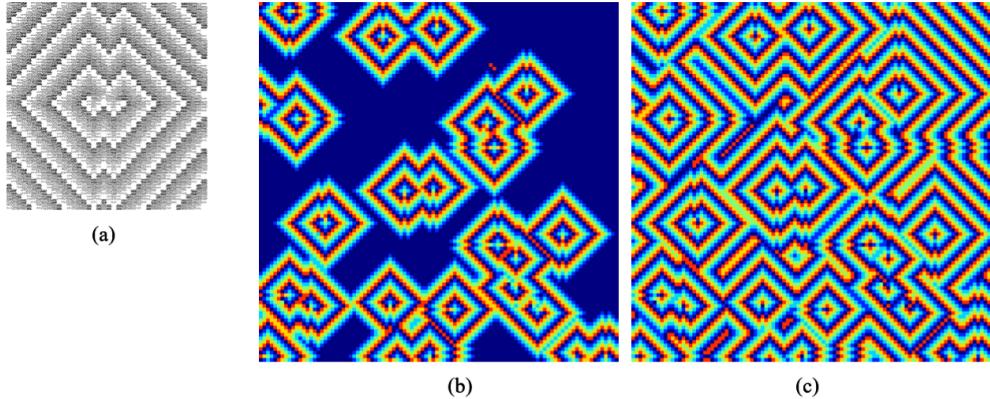


Figure 1: (a) Simulation results of the "Spiral" wave pattern as presented in the original paper. The simulation considers 6 states, where white corresponds to state 0 and dark grey corresponds to state 5. (b),(c) Snapshots of the same simulation on a 300 by 300 grid, illustrating the "Spiral" wave pattern with 6 states. Colors are assigned based on the jet colormap, ranging from 0 to 5 to represent different state values.

There are many modifications of these rules which can be found in the works of Gerhardt *et al*[2], and Markus and Hess [3]. By allowing each state to represent inhibitors or activators, a large set of patterns can be created. All these modifications keep a similar mechanism; a recovery period, excitation of neighbors, and a threshold lead to spiral wave formation.

## 2.2 Host-parasite model

By discretizing a host-parasitoid model and placing the system on a grid, the form is reminiscent of the active CA. This system has nine states,  $A, B, \dots, I$ , and each state goes to the next state ( $I \rightarrow A$ ) with two exceptions. State  $A$  goes to  $B$  only if there is **one**  $B$  neighbor. State  $D$  goes to  $E$  only if there is **one**  $F$  neighbor.  $B$  and  $F$  represent a mature host colony and a mature parasitoid, respectively.

Fig.2 depicts the authors and our own output for this automata at a particular time; white is the absence of organisms, light stippling is the host, and darker stippling represents parasites.

$$\begin{aligned}
& 0 \not\rightarrow 1 \rightarrow 2 \rightarrow 3 \not\rightarrow 4 \rightarrow 5 \rightarrow 6 \rightarrow 7 \rightarrow 8 \rightarrow 0, \\
& 0 \rightarrow 1 \text{ if } N_1 == 1, \\
& 3 \rightarrow 4 \text{ if } N_5 == 1.
\end{aligned} \tag{3}$$

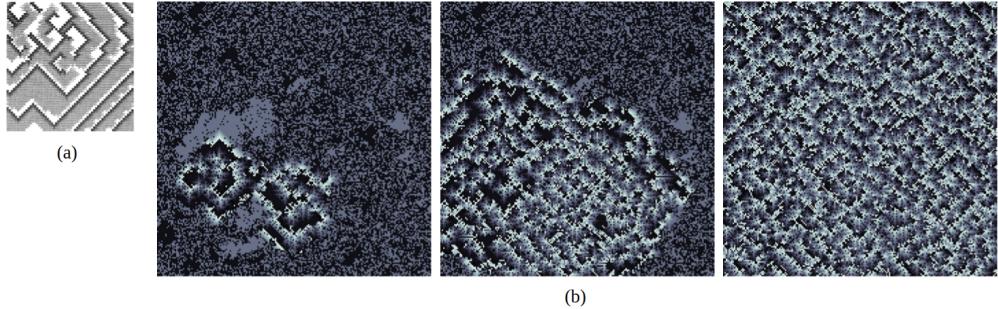


Figure 2: (a) Simulation results of the Host-Parasitoid automata as presented in the original paper. In this representation, state 0 is denoted by white, while state 8 is represented by dark grey. (b) Snapshots of our reproduced simulation of the Host-Parasitoid automata on a 300 by 300 grid illustrate the dynamic interactions between the host and parasite. The colors are mapped using the bone-color map, ranging from the lowest state 0 to the highest state 15. The varying colors capture the evolving nature of the interactions between the host and parasite in the simulated system.

### 2.3 Predator-prey model

The best-known heuristic automata are active media seen in section 2.1, however, predator-prey models can be derived from these types of CA. A fish-shark model is considered. The model consists of an automaton with eight states: (0) -empty sites, (1) -prey, (2)-(6) -predator stages(i-v), and (7) -reproducing predator. The prey reproduces to fill the nearest neighbor's empty sites. Reproducing predators fill sites that have prey or that are empty. The offspring are then stage (i) predators and the parent "dies", leaving an empty site. Prey adjacent to predators disappear ("eaten"). Predators require adjacent prey to consume in order to mature into the next stage, if there are no adjacent spots with prey the predator "dies". This system can be written as

$$\begin{aligned}
& 0 \rightarrow 2 \text{ if } N_7 > 0, \text{ else } 0 \rightarrow 1 \text{ if } N_1 > 0, \\
& 1 \rightarrow 2 \text{ if } N_7 > 0, \\
& (2, 4, 6) \rightarrow 0 \text{ if } N_1 = 0, \text{ else shark moves to next stage}, \\
& (3, 5) \rightarrow (4, 6), \\
& 7 \rightarrow 2 \text{ if } N_1 > 0.
\end{aligned} \tag{4}$$

The author runs simulations on a 50x50 grid with periodic boundary conditions, randomly distributed prey, and one or more initial predators. The complex spatial patterns created undergo complex oscillations of a period of roughly five time steps. We ran a similar simulation which can be compared with the authors in Fig.3. The parameters can be seen in the figure caption. The resulting oscillations seem to be in agreement with results found by Hassel *et al.* [4]

### 2.4 Cellular Automata From More Complex Models

Translating between continuous to discrete domains means creating a rule table that allows for CA to be used and simulated. Two complicated phenomena explored are reaction-diffusion and immunology models.

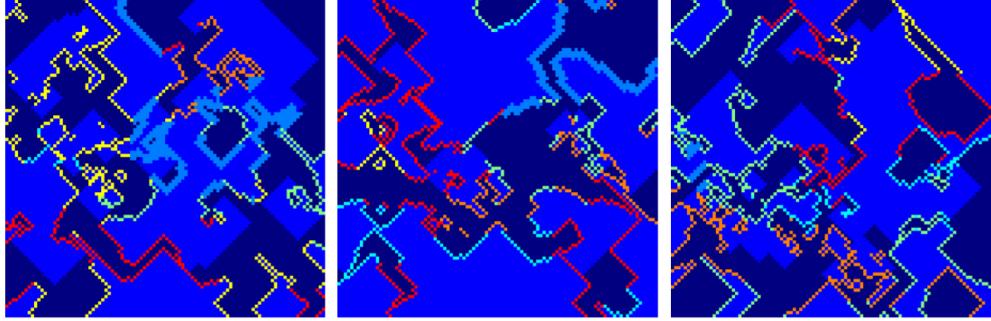


Figure 3: Snapshots of the temporal Predator automata simulation, depicting 8 stages: 0 (empty), 1 (fish), 2-6 (sharks), and 7 (mother shark). Each distinct stage is represented by a unique color according to the jet colormap. The sequence of images captures the temporal evolution and interactions within the Predator automata system.

#### 2.4.1 A reaction-diffusion system

The two variable reaction-diffusion equations in a plane can be given as:

$$\begin{aligned} u_t &= f(u, v) + d_1 \Delta^2 u \\ v_t &= g(u, v) + d_2 \Delta^2 v. \end{aligned} \quad (5)$$

Here,  $u, v$  might represent chemical concentrations, populations of animals, or electrical activity of cells, and  $f, g$  represents the interactions between these two components. The spatial interaction is via diffusion. For the sake of time and patience, given that we already have done a project discretizing this system, we will refer the reader to Alan M. Turing's paper [5] and our midterm project.

#### 2.4.2 An immunology model

So far all the examples have considered the interaction of cells with *nearest* neighbors. However, a continuous state for shape-space simulations of the immune system has been discretized with cell interactions **not** with neighboring sites, but rather, with the complementary sites. This means that if the system is on a  $n \times n$  grid, then the interactions of the cell  $(i, j)$  depend on  $(i', j') \equiv (n - i, n - j)$ . From a model proposed by DeBoer *et al* [6], we arrive at a relatively simple 16-state model for the number of "B-cells",  $b_{i,j}$  at point  $(i, j)$  in state space:

$$b_{i,j}^{t+1} = \begin{cases} \min(b_{i,j}^t + 1, 15) & \text{if } \theta_1 \leq h'_{i,j} \leq \theta_2 \\ \max(b_{i,j}^t - 1, 0) & \text{otherwise} \end{cases} \quad (6)$$

where,

$$h'_{i,j} = w_0 b_{i',j'}^t + w_1 (N' + S' + E' + W') + w_2 (NE' + NW' + SE' + SW'). \quad (7)$$

Here,  $N^t$  means the northern neighbor of  $b_{i',j'}^t$  at time  $t$  and  $w_0, w_1, w_3, \theta_1$ , and  $\theta_2$  are non-negative integers. The variable  $h$  characterizes the total stimulation of each population of B cells which comes from the complementary "shape",  $(n - i, n - j)$ . The purpose of taking the maximum and minimum values is to apply a piece-wise linear approximation to a saturating non-linearity. Figure 4 shows our and the author's results in simulating this automaton. Parameters are given in the legend of the figure. The idea of this model is to mimic the interactions of B cells in a simple two-dimensional shape space. It is important to note that the two dimensions of the model do not correspond to real space but rather some abstract properties (e.g. the shape of the antibody) characterizing B cells. For details on the model and its biological justification see Deboer *et al.* [6]

## 2.5 Developmental Biology

In developmental biology, a simple model that uses the concept of lateral inhibition and thresholds can create spatial two-dimensional patterns mimicking animal coat patterns. This is derived by Young [7]

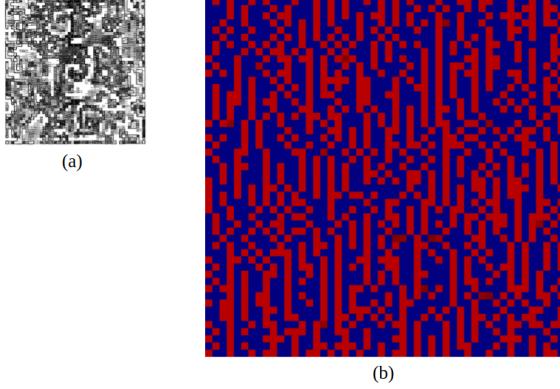


Figure 4: (a) This figure is from the original paper, illustrating a slowly varying spatial pattern for the immune model. In this representation, state 0 is depicted as white, while state 1 is represented as black. (b) Our reproduced simulation output of the same immune model showcases a spatial pattern where dark blue corresponds to state 0, and red corresponds to state 1. The color gradient visually represents the evolving dynamics of the immune model in the simulated environment.

who built his work off Turing [5]. In his rules, cells are in one of two states; the new state is determined by spatially averaging over a circular neighborhood and applying a threshold function:

$$C'_{i,j} = H(\sum_{(i',j') \in N} \mathbf{W}(i - i', j - j') C_{i',j'}), \quad (8)$$

where  $\mathbf{W}$  is a symmetric matrix determined by a radial function,  $N$  is a circular neighborhood, and  $H(u)$  is the Heaviside function.  $\mathbf{W}$  and  $H$  are defined as

$$\mathbf{W} = \begin{cases} W_{\text{inner}} & \text{if } (i - i')^2 + (j - j')^2 \equiv R \leq r_{\text{inner}} \\ W_{\text{mid}} & \text{if } r_{\text{inner}} < R \leq r_{\text{outer}} \\ 0 & \text{if } R > r_{\text{outer}}, \end{cases} \quad (9)$$

where the  $W$ 's and  $r$ 's are free parameters, and

$$H(u) = \begin{cases} 1 & \text{if } u > 0 \\ 0 & \text{if } u < 0. \end{cases} \quad (10)$$

In Fig.5, we show the author's implementation of Young's model for two different weight functions as well as our own implementation. Spots and stripes occur as "inhibition" is varied. This CA has been shown to model the coat patterns of fish, reptiles, and mammals which is possible by varying the weights, the threshold, and the size of the neighborhood. Further reading can be found in the works of Bodenstein [8], Mostow [9], and Nijhout *et al* [10].

### 3 Pattern Adaption for Developmental Biology

We were interested in the CA model used in developmental biology given by equation 8 which we'll show again written in tensor notation

$$c_{t+1}^{(i,j)} = H(\mathbf{W} : \mathbf{C}_t^{(i,j)}) \quad (11)$$

where  $\mathbf{C}_t^{(i,j)}$  is the tensor (same dimension as  $\mathbf{W}$ ) that contains the 'concentration' values of the cells around  $(i,j)$ .  $c_{t+1}^{(i,j)}$  is the 'concentration' value at  $(i,j)$  at the next time step.

We were curious about the effect of the tensor  $\mathbf{W}$  on the final pattern formation. Depending on the matrix parameters, this simple radial function can produce patterns varying from zebra stripes to leopard spots. Our interest lies in reversing this procedure. Is there a method to take an image and calculate the weight matrix?

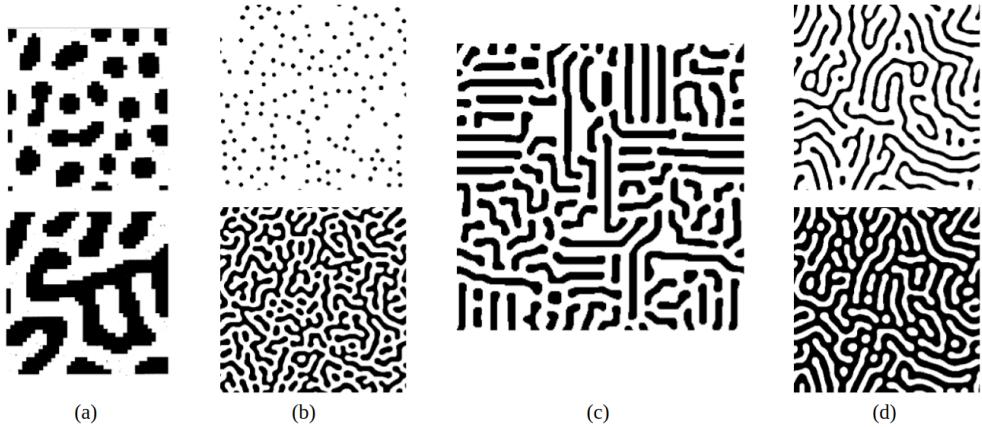


Figure 5: (a) Steady-state solutions to Young’s model as presented in the original paper. Both patterns are formed by maintaining  $r_{inner} = 5$ ,  $r_{outer} = 36$ , and  $W_{inner} = 1$ .  $W_{mid} = -0.34$ ,  $W_{mid} = -0.24$  upper and lower. (b-d) Results from our reproduced simulations on a larger 300 by 300 grid while keeping  $r_{inner} = 4$  and  $W_{inner} = 1$ : (b) Variation of  $W_{outer}$  from -0.15 to -0.32, forming a pattern from top to bottom. The constant parameters are  $r_{outer} = 12$ . (c) Striped pattern formed at  $r_{outer} = 15$  with  $W_{outer} = -0.24$ . (d) Patterns are formed by keeping  $W_{outer}$  constant at -0.34 and varying  $r_{outer}$  from 10 to 15, creating a pattern from top to bottom in a continuum.

We turn our attention to convolution methods which is a powerful tool used for pattern recognition in image processing. Convolution represents the blending of two functions to create a third function. Typically, in image processing a unique matrix known as the convolution matrix is selected that is applied to an image, whose pixel data is stored in a matrix, to extract and manipulate information from the image. Depending on the convolution matrix used, one can saturate, sharpen, enhance, and find the edges of an image. Notice that our model works similarly. Our convolution matrix is  $\mathbf{W}$  and the image is the binary data of our cell array. In our case, we are applying a convolution matrix that is then inserted into a step function to produce a specific pattern. Unlike image processing, where you take the convolution several times to meet a desired end, our model continues to take the convolution at each time step until the system reaches equilibrium. We say the system is stable when the convolution of the image returns the same image.

Using this criterion for stability, are we able to find the convolution matrix for an animal’s coat pattern by simply inputting the desired pattern? Could the convolution method provide insight into how certain animals can change coat patterns rapidly to match their surroundings? Or how animals evolved to create such patterns?

### 3.1 Method

We used images of a zebra, leopard, and giraffe’s coat patterns to build our model. Our processing for obtaining the weight matrix was done by minimizing the amount of error between our initial guess of  $\mathbf{W}$  that produced an image  $\mathbf{I}_W$  and the desired image  $\mathbf{I}$ .

$$\mathbf{W} = \text{argmin} ||\mathbf{I}_W - \mathbf{I}||. \quad (12)$$

We investigated two approaches to calculating  $\mathbf{W}$ .

1. No constraints on the indices of  $\mathbf{W}$  were placed, giving us  $n \times n$  parameters to minimize over.
2. Keeping the radial symmetry of the original weight matrix, we minimized over varying radial functions used to calculate  $\mathbf{W}$ ’s indices

### 3.2 Free index minimization

In this method, we consider the values at all indices of the weight matrix as variables. By initializing the weight matrix with a random seed, we undertake optimization to minimize the difference between

the target pattern and its convoluted version using the weight matrix. Due to the highly nonlinear and nonconvex nature of this difference, traditional gradient-based optimization algorithms prove ineffective. Consequently, we employ a method inspired by Nelder-Mead. The results obtained utilizing this approach are showcased in Fig.6.

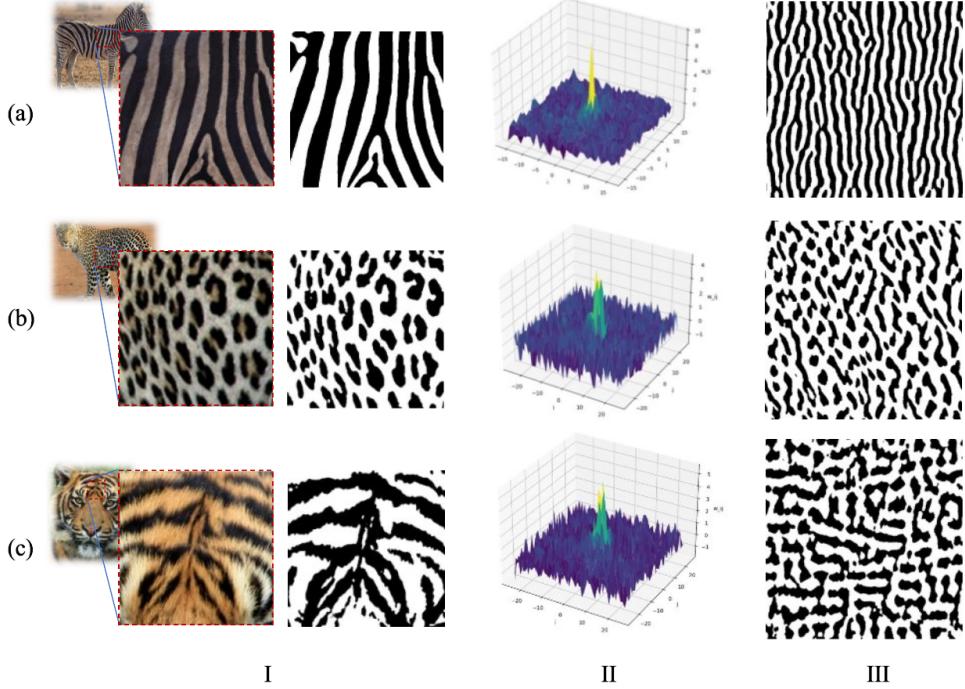


Figure 6: Results of the Free Indices Method applied to three distinct coat patterns—zebra (a), leopard (b), and tiger (c). In column I, a square snippet is displayed, showcasing the original image and its processed version used to generate a binary image. Column II shows a 3D plot of the weight tensor obtained through the Free Indices Method. In column III, the reproduced coat patterns utilizing the obtained weight tensor are presented.

### 3.3 Radial function minimization

Upon observing the results obtained from the Free Indices Method, it becomes evident that weight tensors exhibit a predominantly radially symmetric structure, characterized by a distinct peak at the center and a decaying nature as one moves away from it. Motivated by this observation, we sought to perform optimization over a function of radial distance from the center while keeping the weight tensor constant.

However, the optimization technique initially employed failed to converge for our specific test cases. Undeterred, we redirected our efforts and applied this method to an exponentially decaying cosine wave. Surprisingly, the outcomes yielded intriguing patterns reminiscent of human fingerprints, as illustrated in Fig.7.

## 4 Discussion

There exist numerous examples of realistic models that can be converted into automata. Striking the right balance between a model's simplicity, biological interpretability, and computational efficiency is crucial. Overly simplistic models may lack biological relevance, while excessively complex ones can impede the exploration of broad parameter space within large spatial domains. Despite the substantial increase in computational power since 1993, reduced-order modeling remains a powerful tool for comprehending intricate phenomena. It enables researchers to delve into the underlying mechanics, avoiding sole reliance on numerical simulations for understanding the system of interest.

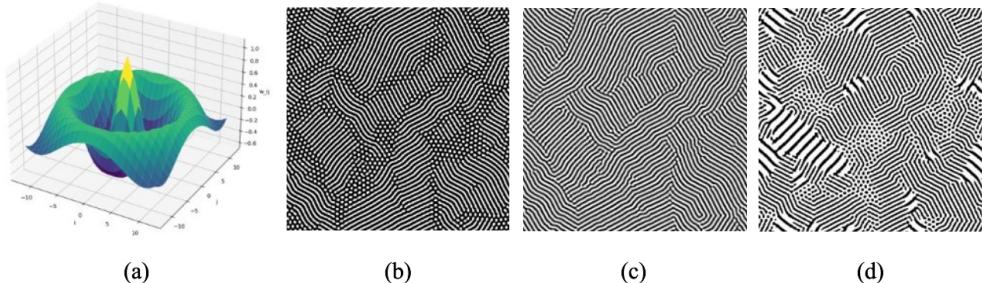


Figure 7: Radially Symmetric Weight Function: This figure presents outcomes from a radially symmetric weight function  $w(r) = 1.1 \cdot \exp(-0.1r) \cdot \cos(0.6r)$ , with a focus on varying  $r_{\max}$ . (a) displays the resulting weight matrix. (b), (c), and (d) exhibit the outcomes for  $r_{\max} = 10$ ,  $r_{\max} = 13$ , and  $r_{\max} = 18$ , respectively.

Cellular Automata (CA) models, known for their speed and ease of implementation, provided us with rapid results within a few hours using Python. The visual feedback they offer exhibits a striking resemblance to experimentally observed patterns. While CA should not replace rigorous mathematical models, they serve as an invaluable initial step in the modeling process. CA facilitates a seamless transition from non-rigorous statements of mechanisms to formal models.

Building upon the presented work, our exploration extended into pattern adaptation, where we devised an analytical method to identify rules governing specific pattern formations by determining the associated weight tensor. Through several examples, we have demonstrated the feasibility of finding the weight tensor for certain biological patterns observed in nature. Further refinement of this method’s efficiency and its application to a broader range of biological data remains a promising avenue for future research.

## 5 Future Directions

### 5.1 Explaining Coat Patterns in an Evolutionary Perspective

One promising avenue for future research involves exploring the potential use of the weight tensor as a convolution kernel to investigate the evolutionary origins of coat patterns. By integrating genetic and environmental factors into the modeling process, we aim to explore how specific patterns may have evolved over time. This approach may allow us to simulate the interplay between genetic mutations, selective pressures, and environmental conditions, potentially providing insights into the adaptive significance and evolutionary trajectories of coat patterns across different species.

### 5.2 Exploring Dynamic Pattern Adaptation

In the realm of dynamic pattern adaptation, the use of the weight tensor as a convolution kernel may offer a potential tool to investigate how patterns adapt over time. By incorporating temporal dynamics into the convolution process, we can model how environmental changes and selective pressures might influence the adaptive responses of biological systems. This approach may enable a more nuanced understanding of the mechanisms governing pattern adaptation in response to evolving conditions.

### 5.3 Compressing Large Datasets of Biological Systems

Efficiently compressing large datasets of biological systems is essential for storage and analysis. The exploration of the weight tensor as a convolution kernel might provide a potential means to distill essential information from complex datasets while preserving the salient features. This convolution-based compression approach has the potential to facilitate the extraction of meaningful patterns from large datasets, contributing to the development of more streamlined and interpretable representations of biological systems.

## References

- [1] E. Bard and E.-K. Leah, “Cellular automata approaches to biological modeling,” *Journal of Theoretical Biology*, vol. 160, pp. 97–133, June 1992.
- [2] M. Gerhardt, H. Schuster, and J. Tyson *Science*, vol. 247, pp. 1563–1466, Nov. 1990.
- [3] M. Markus and B. Hess *Nature, Lond*, vol. 437, pp. 56–58, 1990.
- [4] P. Hassell, M. H. Comins, and M. R.M. *Nature, Lond*, vol. 353, pp. 255–258, 1991.
- [5] A. Turing, “The chemical basis of morphogenesis,” *Philosophical Transactions of the Royal Society of London. Series B, Biological Sciences*, vol. 237, no. 641, pp. 37–72, 1952.
- [6] R. Deboer, L. Segel, and P. A.S. vol. 19, 1991.
- [7] D. Young *Math, Biosci*, vol. 72, pp. 51–58, 1984.
- [8] L. Bodenstein *Cell Diff*, vol. 19, pp. 19–33, 1986.
- [9] G. Mostow, “Mathematical models for cell rearrangement,” vol. 19, 1975.
- [10] H. Nijhout, G. Wray, C. Krema, and C. Teragawa *Syst. Zool.*, vol. 35, pp. 445–457, 1986.