entropy

MDPI

Article

Minimal Developmental Computation: A Causal Network Approach to Understand Morphogenetic Pattern Formation

Santosh Manicka and Michael Levin *

Allen Discovery Center, Tufts University, Medford, MA 02155, USA; santosh.manicka@tufts.edu
* Correspondence: michael.levin@tufts.edu

Abstract: What information-processing strategies and general principles are sufficient to enable self-organized morphogenesis in embryogenesis and regeneration? We designed and analyzed a minimal model of self-scaling axial patterning consisting of a cellular network that develops activity patterns within implicitly set bounds. The properties of the cells are determined by internal 'genetic' networks with an architecture shared across all cells. We used machine-learning to identify models that enable this virtual mini-embryo to pattern a typical axial gradient while simultaneously sensing the set boundaries within which to develop it from homogeneous conditions—a setting that captures the essence of early embryogenesis. Interestingly, the model revealed several features (such as planar polarity and regenerative re-scaling capacity) for which it was not directly selected, showing how these common biological design principles can emerge as a consequence of simple patterning modes. A novel "causal network" analysis of the best model furthermore revealed that the originally symmetric model dynamically integrates into intercellular causal networks characterized by broken-symmetry, long-range influence and modularity, offering an interpretable macroscale-circuit-based explanation for phenotypic patterning. This work shows how computation could occur in biological development and how machine learning approaches can generate hypotheses and deepen our understanding of how featureless tissues might develop sophisticated patterns—an essential step towards predictive control of morphogenesis in regenerative medicine or synthetic bioengineering contexts. The tools developed here also have the potential to benefit machine learning via new forms of backpropagation and by leveraging the novel distributed self-representation mechanisms to improve robustness and generalization.

Keywords: biological computation; developmental patterning; distributed information processing; collective phenomena; causal information flow; biological circuits; artificial embryogeny

check for updates

Citation: Manicka, S.; Levin, M. Minimal Developmental Computation: A Causal Network Approach to Understand Morphogenetic Pattern Formation. *Entropy* **2022**, *24*, 107. https://doi.org/10.3390/e24010107

Academic Editors: David Wolpert and Jessica Flack

Received: 7 September 2021
Accepted: 7 January 2022
Published: 10 January 2022

1. Introduction

How does a developing embryo self-organize into a patterned structure arranging the various differentiated morphological features using input information from its own cells [1,2]? In essence, how does an embryo compute its own pattern [3–9]? Morphogenesis, whether embryonic or regenerative, is an intriguing paradigm for computer science, in addition to biology and biomedicine, because it provides proof-of-principles of a dynamic, strongly embodied computational architecture [10–12]. The results of its computations alter the architecture and structure of the computing medium in real-time. Morphogenesis is extremely robust (reviewed in [13]), partly because it has evolved to maintain reliable signaling and morphological decision-making while it itself is actively remodeling. While progress in molecular genetics has made huge strides in uncovering the mechanisms *necessary* for morphogenesis, there are still significant gaps in a mature understanding of the strategies and meso-scale design principles *sufficient* to enable the right combination of reliability and plasticity [14–16]. How do cellular collectives make decisions with respect to signaling and cell behavior that reliably result in the correct species-specific target morphology?

To answer such questions, various conceptual and computational models have been developed [17–24]. Lewis Wolpert, a pioneer in this field, introduced the concept of "positional information", where a morphogen gradient develops along an axis that the cells could use to decode their relative positions and make morphogenetic decisions in embryogenesis or regeneration [25]. He used the famed metaphor of the 'French-flag' to characterize a gradient-like pattern and later proposed models for how such patterns could be developed, sustained and even regenerated [25–27]. Pattern formation is a dynamical and multifaceted phenomenon that is not restricted to embryogenesis. For instance, a variety of biological systems possess capabilities of regenerative patterning, that is, adult forms can regenerate full patterns from sub-regions when severed [28]. To understand patterning in its various forms, therefore, several conceptual and mathematical models have been developed since Wolpert's work. A few notable concepts include "organizers", morphogenetic fields, epimorphosis, morphallaxis and intercalation [20,29,30]. The concept of intercalation, for instance, explains how an arbitrary piece of a planarian regenerates the whole worm; the anterior and posterior poles of the stump have non-adjacent "positional values" relative to their neighbors, the continuity of which is then gradually restored leading to the formation of the full original pattern. Most notable mathematical models of patterning are based on either simple diffusion or the more complicated reaction-diffusion mechanisms [19,21,22,24], with some exceptions like the "clock and wavefront model" [31]. A model introduced by Alan Turing, known as "Turing patterns", for instance, is based on a mechanism of "short-range activation and long-range inhibition" driven by inherent dynamical instabilities [32]. A variety of computational models that view cells as discrete entities ("agents") containing memory and performing actions or use artificial neural networks in the background for morphogenetic decision-making have been attempted as well [33–38]. Even though these models explain certain aspects of patterning in concise and intuitive ways, they do not reveal the full underlying dynamical richness of this nonlinear phenomenon. For instance, a typical analysis of Turing patterns predicts the size and wavelength of the emergent patterns using mathematical expressions involving the ratio of the activator and inhibitor diffusion coefficients obtained by linearizing the model. Even though this approach is useful, it still lacks the full picture of a *nonlinear computational* account of patterning that moreover is also interpretable. Such an account may be invaluable for systematically addressing the inverse patterning problem where one wishes to modify the underlying regulatory systems to achieve desired patterning outcomes [13,39].

One of the central challenges with respect to models is thus an understanding of the high-level information-processing principles they leverage towards the patterning outcomes. For instance, the "balancing model" proposed by Wolpert represents cells as point-entities that make linear measurements such as computing simple averages. As helpful as the model is in hypothesizing high-level principles, they do not elucidate the underlying biological complexity. Models based on reaction-diffusion or artificial neural networks, on the other hand, capture some of the biological complexity but suffer from the drawback that it is difficult to elucidate the underlying nonlinear causal relationships using conventional analysis techniques [40,41].

Information theory offers tools [42–44] to overcome some of the above-mentioned barriers by inferring the complex causal relationships among the components of a system from timeseries data using a probabilistic perspective [45], although not without challenges [46]. In the same way, dynamical systems offer tools to infer "dynamic causal models" (DCM) [47] that elucidate the causal relationships underlying a dynamical phenomenon from a deterministic (non-probabilistic) perspective [45]. These efforts to explicate the information-processing mechanisms of complex dynamical models have given birth to a new field dedicated to solving the problem of "interpretability" [48]. As a representative example, the method of "saliency map" elucidates the parts of an input image that an output neuron of an image-recognizing ANN is most sensitive to by computing the mathematical derivative of the output with respect to the input [49,50], thereby revealing the character of that neuron in a human-interpretable way. Although saliency map and its variants [50] have been used to characterize the high-level information-processing principles of ANNs, they have not been used to similarly analyze non-neural biological models, to the best of our knowledge. With the advent of differentiable programming [51], the time is ripe for the extension of those methods to non-neural recurrent dynamical models. In summary, the limitations of conventional analysis methods' inability to reveal effective information-processing principles can now be surpassed by extending the techniques developed for interpreting ANNs. Besides, we can now develop methods to more closely link computation in systems whose outputs modify the computing medium and use machine learning to discover morphogenetic control strategies that serve as hypotheses for developmental biology or design policies for synthetic bioengineering.

The goal of this paper is thus two-fold: (1) to design a minimal nonlinear dynamical model of generic pattern formation using plausible biological priors; and (2) to characterize the high-level organizational principles that the model employs by applying analysis methods that are extensions of conventional methods used for ANNs.

2. Model and Methods

We designed a minimal model of bounded axial patterning—a phenomenon that commonly occurs during embryogenesis and regeneration where a featureless tissue develops an epidermis (boundary) and an axis within it (e.g., anterior-posterior, dorsal-ventral, etc.) along which distinct morphological features later appear (Figure 1). Rather than simulating any one species' morphogenesis, we seek to understand axial polarity, a generic mechanism used throughout biology [52–60].

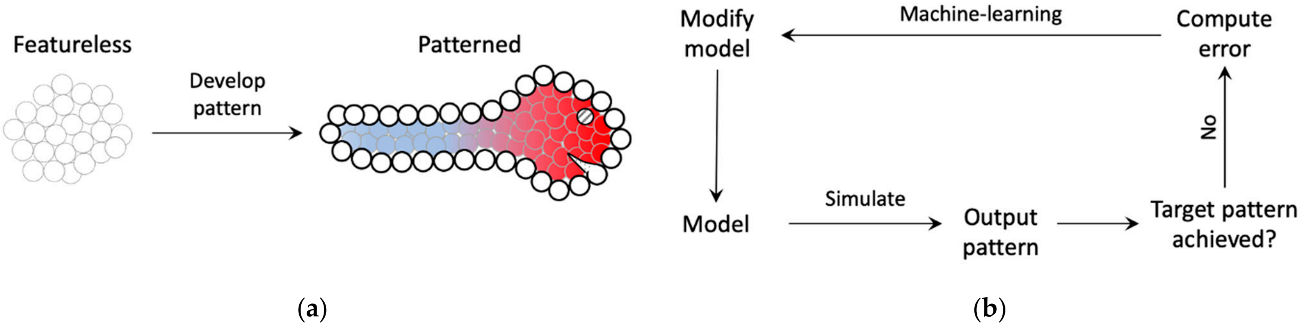


Figure 1. Schematic of the modeled biological phenomenon and how machine-learning is used to design the model. (**a**) During early embryogenesis, a relatively featureless embryo develops distinct axial patterns along with a distinct outer layer known as the epidermis (figure of larva was inspired by [1] (p. 11)). The distinct colors represent the origin of the differentiation of the embryo into distinct morphological features. The cells on the boundary (thick empty circles) represent the epidermis. (**b**) A model with unknown parameters is trained using machine-learning, that uses gradient-descent-like methods to 'backpropagate' the error between the observed and target patterns to the model parameters, to produce the desired pattern.

The functional output sought in this model is to develop an axial pattern of positional information at the same time marking the boundary (epidermis) within which to develop it (Figure 1). Even though the model is simplified, it captures certain biological priors that orchestrate embryogenesis, such as intra-cellular network control (e.g., genetic, biochemical signaling) with features supporting the implementation of planar cell polarity, local inter-cellular connectivity, and bounded development. The following are the main features of the model (Figure 2). It consists of a bounded (finite) linear chain of cells, each characterized by a 'cell type', and connected by edges representing gap-junctions [61] that allow them to communicate physiologically. Every cell, furthermore, contains a self-regulating connection representing autocrine signaling in biological cells that gives them a capacity to regulate themselves [62] (p. 238). The cell type, along with the gap junctions and self-regulating connections, dictates the cell's behavior (response to inputs). The dynamical state (output) of a cell is referred to as its "activity level" that is steered by its interactions with the neighboring (input) cells' activity levels; this represents the cell's emergent positional information and identity within the tissue. The connection weights and the cell types are themselves dynamic variables whose properties are determined by an "intrinsic signaling controller" that each cell possesses internally with an architecture shared across all cells, representing the generic signaling networks that biological cells possess (e.g., genetic networks, biochemical signaling networks, etc.). The dynamic nature of the gap-junctions represents the dynamically changing permeability of their biological counterparts due to voltage-gating, for example [63]. Furthermore, since a gap-junction connects two adjacent cells, its dynamic weight is jointly determined by the cells' intrinsic controllers. Every cell also possesses a "boundary-marker" state, a dynamical variable representing the extent to which the cell is deemed to lie at the boundary—the higher the value of the state, the larger is the cell deemed to be at the boundary. Like the activity state, the boundary-marker state of a cell depends on the boundary-marker states of the neighboring cells as well as itself. These boundary-marking signals are generated by a second intracellular controller known as the "boundary signaling controller". The function of the boundary-marker is to dampen all the activities of a cell to a degree proportional to the level of the boundary-marker (Figure 3)—the more a cell tends to be a boundary cell, the more dampened its overall activity is, including the patterning state, cell properties and the internal controller states. This feature represents the biological setting where the embryonic epidermis is relatively less active compared to the inner cells. In this regard, the boundary-marker represents the antagonistic version of "growth factors" that biological cells signal to neighboring cells to divide [64]. Another important feature of the model is that the intrinsic controller is designed to support the implementation of planar cell polarity (PCP) that gives the cell a sense of polarity, a feature that is known to play an important role in the development of organism-level axial polarity [65,66]. In the model, specifically, the anterior (left) column of the intrinsic controller influences the anterior gap-junction weight, whereas the posterior (right) column influences the posterior gap-junction weight (Figure 2). This arrangement allows machine-learning to assign distinct modes of behavior to the two polar columns, thereby effectively functioning as PCP (described in more detail below). Lastly, the fact that every cell in the model possesses two separate internal controllers tasked with unique functionalities represents the biological analogue of multiple signaling networks working in concert inside a cell, representing for example the biochemical and bioelectrical control systems [67].

In summary, while the cells communicate over the gap-junctions to form the target pattern within a boundary, their internal signaling networks help build the tissue-level intercellular network and the network-boundary themselves.

The formal definition of the model is depicted in Figure 3. The model is defined by a set of recurrent ordinary differential equations (ODE) that describe how the variables of the model, namely, the activity states (s), cell types (p), gap-junction weights (j_g), self-weights (j_s), boundary-marker states (b), intrinsic controller states (r) and the boundary-marker states (y) interact with each other. These variables are ultimately governed by a set of parameters (shown in red in Figure 3) that are trained by machine-learning. We used a specific machine-learning method called backpropagation through time (BPTT) to identify appropriate model parameters that solve the patterning problem. This system models the following aspects of biology. The machine learning loop represents the role of the evolutionary process, which selects for effective morphogenesis by providing cells with a genome encoding the correctly parametrized regulatory network. As occurs in biological evolution, this parameterization process is external to the lifetime of an individual creature and its genomically specified network. In turn, the specific networks produced by the training algorithm represent the results of genomes—the operation of individuals during the morphogenesis of their embryonic development.

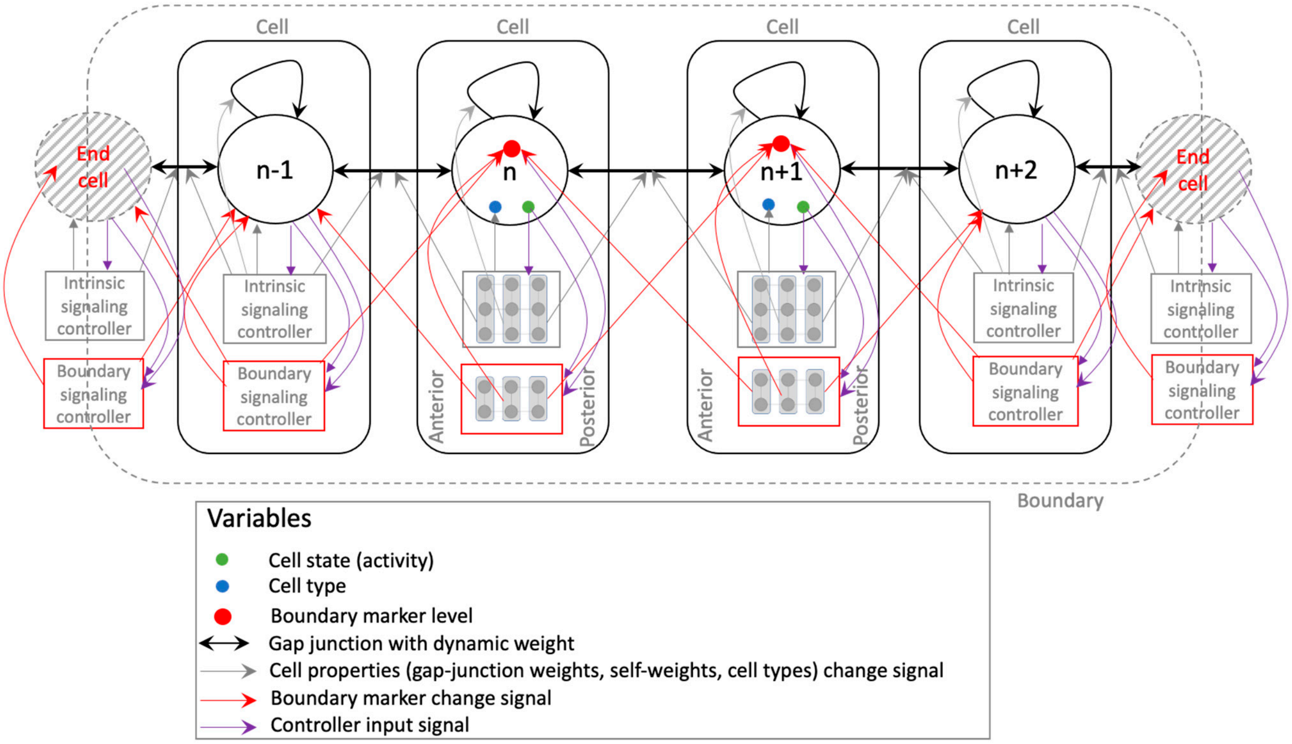
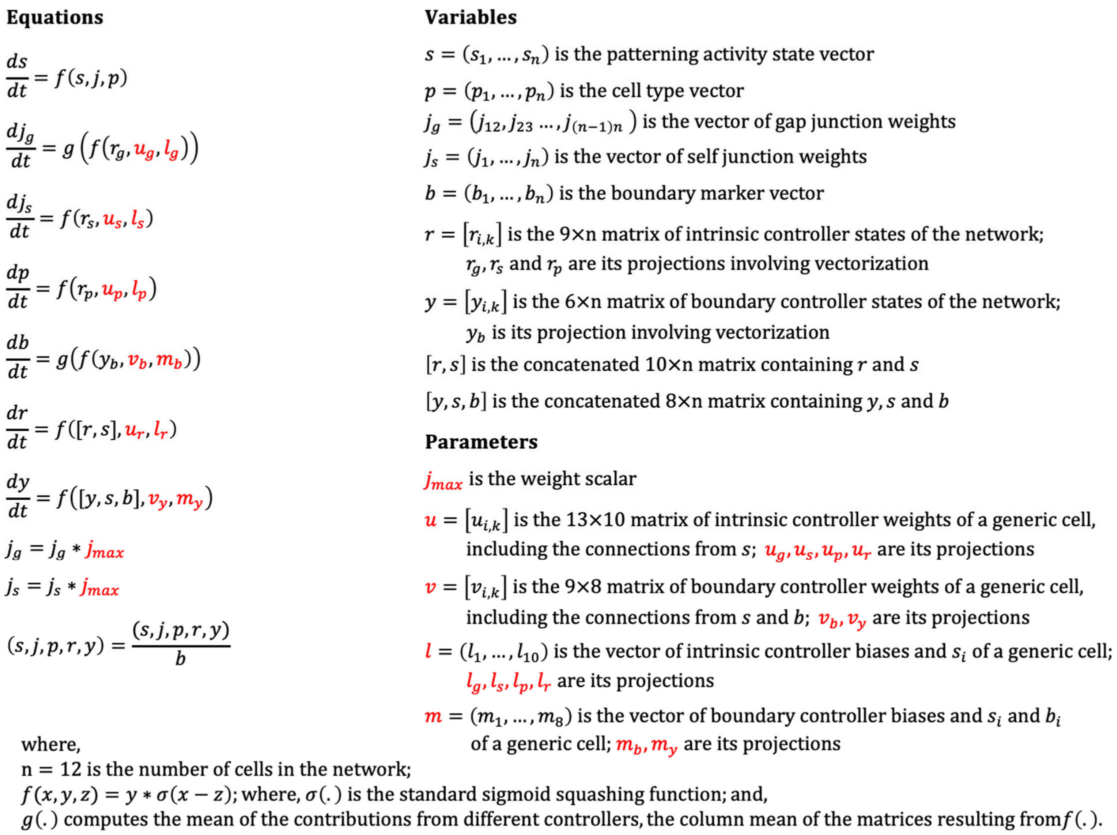

Cell
Cell
Cell
Cell
End cell
n-1
n
n+1
n+2
End cell
Intrinsic signaling controller
Boundary signaling controller
Intrinsic signaling controller
Boundary signaling controller
Anterior
Posterior
Anterior
Posterior
Intrinsic signaling controller
Boundary signaling controller
Intrinsic signaling controller
Boundary signaling controller
Boundary
Variables
● Cell state (activity)
● Cell type
● Boundary marker level
⟷ Gap junction with dynamic weight
→ Cell properties (gap-junction weights, self-weights, cell types) change signal
→ Boundary marker change signal
→ Controller input signal

Figure 2. Schematic of the model of axial pattern development. The model comprises of a finite linear chain of cells (a total of 12 cells comprise the model used in this work). We hypothesize that the following elements will be sufficient to implement emergent axial patterning. Every cell has two kinds of signaling networks, one that determines the properties of the cell, namely the cell type and the gap-junction weights, and the other that signals to the cell the extent to which it lies at the boundary. These two factors are expected to act in concert in simultaneously detecting the boundaries and developing a gradient-like phenotypic pattern within the detected boundaries where the pattern would be expected to taper off. The intrinsic controller is a 3 × 3 lattice (totaling 9 nodes), and the boundary controller is a 2 × 3 lattice (totaling 6 nodes), depicted as part of the representative cells n and $(n+1)$ in the middle. As the connections indicate, the anterior and posterior (1st and 3rd) columns of the intrinsic controller influence the anterior and posterior gap-junction weights respectively, and the central column influences the cell's self-weight. In the same way, the most anterior and posterior (1st and 3rd) columns of the boundary controller signal to the anterior and posterior cells, respectively. Finally, in each cell, the cell state influences all the nodes of either controller, the cell type is influenced by all of the intrinsic controller nodes, and all of the boundary controller nodes are influenced by the cell's own boundary-marker level.

The specific problem that the model learned to solve comprises of two main features (details below): (1) a network activity state pattern that is shaped like a simple gradient, with positive values at the anterior and negative values at the posterior and tapering off (goes to zero) at either pole; and (2) a boundary marker pattern where the boundary cells have double the value compared to the cells in between (a value of 2 was set as the target for the stop cells and 1 for the rest). These target patterns were meant to serve as idealized axial polarity and boundary marker patterns to encourage the model to develop patterns that approximately match them.

The method of BPTT broadly involves instantiating a model with random parameters chosen from a specific range, simulating the model for a certain number of steps, then calculating the loss as the difference in the observed network activity and boundary marker patterns with the corresponding set targets and finally backpropagating the loss back to the parameters. The loss function involved a simple mean squared error (MSE) between the observed and the target patterns with equal weights assigned to the network-activity and the boundary-marker patterns. This process was repeated (over several thousands of iterations) until a model with a satisfactory performance was obtained. The parameters u, v, l and m were initialized in the interval $[-1, 1]$, and the parameter j_{max} in the interval $[1, 2]$. The initial conditions of all the variables were set to 0 during the training. We utilized the software package *Pytorch* [68], that employs automatic differentiation techniques, to implement BPTT.



Equations

$$\frac{ds}{dt} = f(s, j, p)$$

$$\frac{dj_g}{dt} = g\left(f(r_g, u_g, l_g)\right)$$

$$\frac{dj_s}{dt} = f(r_s, u_s, l_s)$$

$$\frac{dp}{dt} = f(r_p, u_p, l_p)$$

$$\frac{db}{dt} = g(f(y_b, v_b, m_b))$$

$$\frac{dr}{dt} = f([r, s], u_r, l_r)$$

$$\frac{dy}{dt} = f([y, s, b], v_y, m_y)$$

$$j_g = j_g * j_{max}$$

$$j_s = j_s * j_{max}$$

$$(s, j, p, r, y) = \frac{(s, j, p, r, y)}{b}$$

Variables

$s = (s_1, \ldots, s_n)$ is the patterning activity state vector

$p = (p_1, \ldots, p_n)$ is the cell type vector

$j_g = (j_{12}, j_{23} \ldots, j_{(n-1)n})$ is the vector of gap junction weights

$j_s = (j_1, \ldots, j_n)$ is the vector of self junction weights

$b = (b_1, \ldots, b_n)$ is the boundary marker vector

$r = [r_{i,k}]$ is the 9×n matrix of intrinsic controller states of the network;
r_g, r_s and r_p are its projections involving vectorization

$y = [y_{i,k}]$ is the 6×n matrix of boundary controller states of the network;
y_b is its projection involving vectorization

$[r, s]$ is the concatenated 10×n matrix containing r and s

$[y, s, b]$ is the concatenated 8×n matrix containing y, s and b

Parameters

j_{max} is the weight scalar

$u = [u_{i,k}]$ is the 13×10 matrix of intrinsic controller weights of a generic cell,
including the connections from s; u_g, u_s, u_p, u_r are its projections

$v = [v_{i,k}]$ is the 9×8 matrix of boundary controller weights of a generic cell,
including the connections from s and b; v_b, v_y are its projections

$l = (l_1, \ldots, l_{10})$ is the vector of intrinsic controller biases and s_i of a generic cell;
l_g, l_s, l_p, l_r are its projections

$m = (m_1, \ldots, m_8)$ is the vector of boundary controller biases and s_i and b_i
of a generic cell; m_b, m_y are its projections

where,
n = 12 is the number of cells in the network;
$f(x, y, z) = y * \sigma(x - z)$; where, $\sigma(.)$ is the standard sigmoid squashing function; and,
$g(.)$ computes the mean of the contributions from different controllers, the column mean of the matrices resulting from $f(.)$.

Figure 3. Formal definition of the model. All variables and parameters are continuous in that they can in principle assume any value on the real number line. The parameters (red) are trained and thus fixed during simulations. All variables are scaled by the boundary marker level b, except itself, representing its dampening effect (akin to a time constant)—the greater the value of b (the greater the cell tends to be a boundary cell), the more dampened its overall activity, including the patterning state, cell properties and internal controller states, is. Subscripted parameters and variables denote unique projections of the corresponding parameters and variables that can be inferred from the connectivity diagram shown in Figure 2. In cases where the value of a variable is determined by inputs from multiple controllers, such as the gap-junctions and the boundary-markers, the averaging operator $g()$ computes the means of those contributions. The case of gap-junction weight updates exemplifies these concepts. The weight of every gap-junction $j_{(n-1)n}$ is determined by the intrinsic controllers of cells $(i-1)$ and i, specifically by the posterior column in the controller of the anterior cell and the anterior column of the posterior cell. Accordingly, the dimension of r_g would be $3 \times 2 \times (n-1)$, where the dimensions 3×2 represents the three nodes each of the two controller columns. This r_g would then be partially vectorized, with the two columns contributing to each gap-junction concatenated into one, yielding a matrix with dimensions $6 \times (n-1)$. In the same way, the dimension of u_g would be 2×6, representing the two gap-junctions each contributed to by one column of the generic cell's intrinsic controller totaling six nodes (three nodes per column). Thus, the multiplication of u_g with r_g yields a matrix of dimensions $2 \times (n-1)$. Finally, the averaging operator $g()$ computes the column-wise mean of the previous matrix resulting in a $1 \times (n-1)$ vector of updates to all the gap-junctions.

We sought to explore potential information-processing strategies and general principles that biological systems may employ for the purpose of developmental patterning. To that end, we answer the following questions in this paper: (1) can the model be trained to solve the patterning problem? and (2) how does a successful model work, specifically, how is the information about the target network activity pattern organized in the intrinsic controllers? To answer the second question, we formulated a measure of causal influence (CI) to quantify a graded measure of information—the more a variable x is said to causally influence a variable y in the context of the model's full state, the more x is deemed to contain information about y in that context. Specifically, we quantify the amount of information contained by variable $x(t)$ about variable $y(t+\tau)$ via a measure of CI at timescale τ, defined as $\partial y(t+\tau)/\partial x(t)$, evaluated in the context of the overall model state at t; the higher the absolute value of this derivative is, the greater is the causal influence. We ascribe the causality of this measure to the necessity and sufficiency of $x(t)$ to impart a change in $y(t+\tau)$ in a local sense; a small change in $x(t)$ in one direction will cause a change in $y(t+\tau)$ in precisely one direction depending on the sign of the derivative. Moreover, CI is directional and asymmetric by definition; it flows from the source x at time t to the target y at time $(t+\tau)$.

In the rest of the paper, we use this formulation of multi-timescale causal influence to quantify the amount of information about patterning activity state $s(t+\tau)$ contained in the intrinsic controller states $r(t)$, defined as the following Jacobian tensor evaluated at the full network state at t:

$$J_s(\tau) = \left[\frac{\partial s_i(t+\tau)}{\partial r_{j,k}(t)}\right]_{i\times j\times k} ;\ i,k = 1,\ldots,n \text{ and } j = 1,\ldots,9 \tag{1}$$

Here, i indicates the influenced cell, j the intrinsic controller node and k the influencing cell. This Jacobian tensor can be coarse-grained and visualized as a causal network that depicts how information is generated, hence organized, at the network level. A causal network of the model is defined as a directed network whose nodes represent variables (e.g., genetic states, patterning activity states, etc.) of the original model and a connection represents a non-zero influence between the connected variables. A unique causal network exists for every possible timescale τ and time t, and for every possible pair of variables X and Y (they could represent the same variable). Due to its association with arbitrary timescales $\tau \geq 1$, a causal network can only be generated via mathematical integration; hence, we refer to this method as "causal network integration". In this sense, our method could be considered as a nonlinear generalization of methods that employ network-structure-based integration techniques, such as eigenvector centrality, that implicitly assume linear dynamics [69,70]. We utilized the software package *Pytorch* [68], built on automatic differentiation techniques, to integrate the model. A discrete version of this method has been developed to integrate Boolean networks and cellular automata [71] (pp. 124–132).

The main difference between CI and most existing measures of information-processing and causality such as transfer entropy, Granger causality, etc. is that CI is a deterministic measure while the others are statistical in nature. This is because CI is computed off a directed acyclic graph (commonly referred to as DAG) representation of the operations in the dynamical model known as the "computational graph" using the chain rule [68], whereas the statistical measures utilize probability distributions that are either observed or inferred from the empirical data [72]. In other words, CI captures causality based on the linkage among the computational operations in a dynamical model, whereas the statistical measures of causality are based on empirical correlations in a dataset. A natural limitation of CI in its present form is that it requires a differentiable model and cannot be computed over data alone.

3. Results

We hypothesized that the biologically inspired minimal dynamical model can solve the patterning problem using self-organization alone, that is, without the aid of any external instructions or special initial conditions. We found that the model can indeed solve this problem in this way. We also found, using the causal network integration approach, that the model dynamically breaks symmetry and integrates into a macroscale network with emergent patterns whose characteristic features explain the shape of the activity pattern. Below, we describe these results in detail. We start with a description of the results of the training and the overt patterning of the best-trained model, followed by a depiction of how the patterning behavior is reflected in the activity of the internal controllers and in single-cells, concluding with a characterization of the causal network machinery that links the controller activity to the overt behavior. The results of the parallel in-depth analysis of the boundary-marker patterning are presented in SI (Figures S2–S8).

3.1. The Model Learns to Generate the Correct Activity Patterns and Mark Boundaries

We simultaneously trained a set of 100 models for about 100,000 iterations, a majority of which (72%) attained satisfying performance with an average MSE loss of about 0.04 and a minimum of about 0.02, compared to the performance of a set of random untrained models with an average MSE loss of about 6.1 (Figure 4), where the ideal loss is 0.0. Thus, machine-learning helped discover hypotheses (coded in the form of models) about how biological systems might solve the patterning problem. In the following sections, we analyze the best-performing model to decipher those hypotheses (information-processing strategies). We chose to analyze a single representative top-performing model because its behavior is qualitatively similar to the average behavior of the top-performing ensemble (Figure S1).

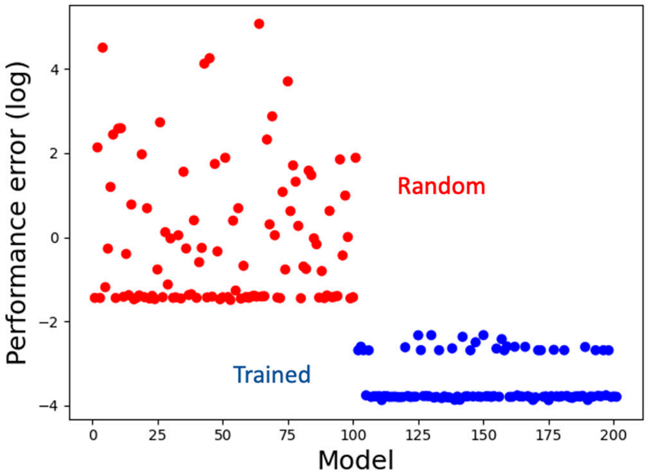
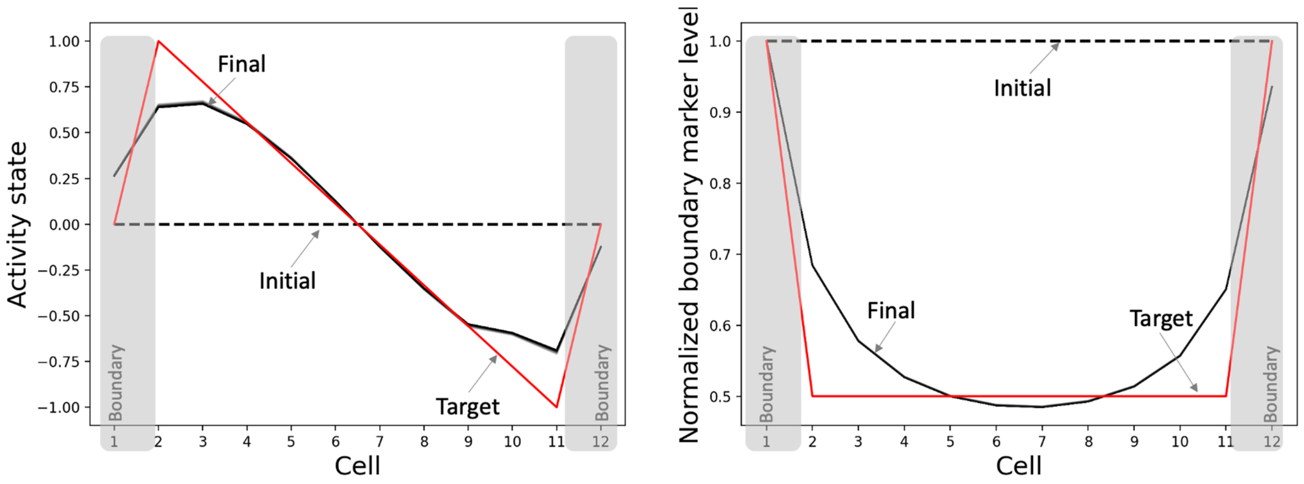


Figure 4. Training performance. Models trained using machine-learning have low performance errors (blue) compared to the random models (red). The line of blue dots at the bottom comprises the set of top-performing models (72% of the trained models) with similar scores.

3.2. Analysis of Cellular Activity and Structural Patterns

3.2.1. The Model Develops Network Activity and Boundary-Marker Patterns Establishing a Correct Axial Gradient Pattern within the Tissue

The model develops the network activity pattern and marks the boundaries starting from homogeneous conditions as expected (Figure 5). This process takes about 4000 time-steps, equaling 40 simulation-seconds, with every second comprising a sequence of 100 synchronized dynamical updates of the model's variables. Specifically, the network develops an axial gradient pattern where the activity states drop mostly smoothly from high positive values to low negative values along the anterior-posterior axis, with the activities of the boundary cells (positions 1 and 12) tapering off towards zero (Figure 5a). In total, this constitutes a 98.4% match with the target pattern relative to the maximum mismatch of the sign-flipped target pattern. The overall (normalized) shape of the boundary-marker pattern, where the boundary cells have the highest levels of the marker and the difference between the marker levels of the adjacent cells is the largest for the boundary cells, matches the target with a score of 96.1% (Figure 5b). We also found that although the normalized pattern persists, the network-activity pattern slowly heads toward a flat shape over a long period of time. This behavior is compatible with biological examples where the positional information patterns persisting through key developmental stages disappear during the maintenance phase of adulthood or under the impact of significant aging.

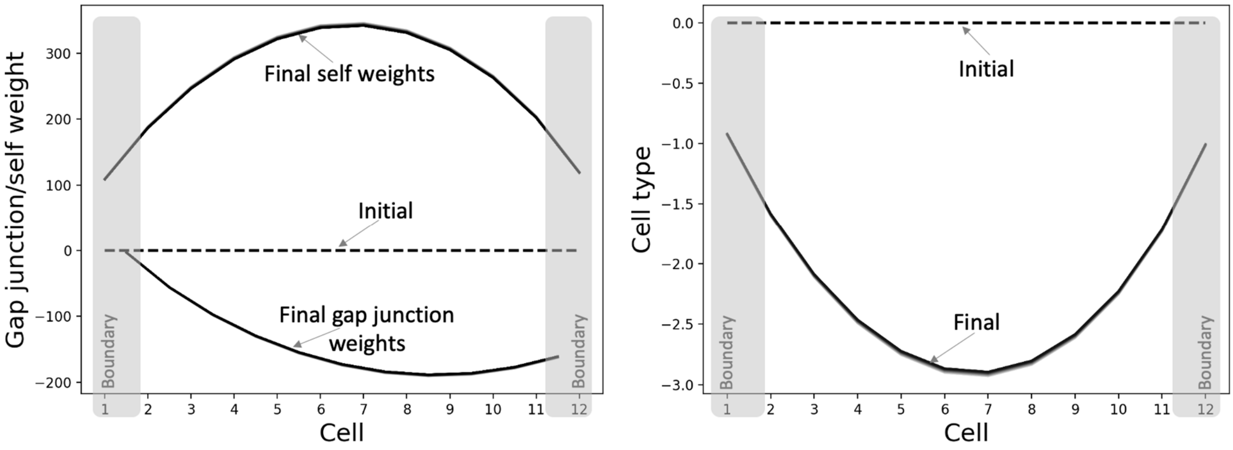


(**a**) (**b**)

Figure 5. Patterning and boundary-marking behavior of the best-performing model. When initialized from homogeneous conditions and run for 4000 time-steps, the (**a**) network activity state pattern converges to a pattern (solid black) that closely matches the target pattern (red), and (**b**) the normalized boundary marker pattern reaches (solid black) a state where the cells at anterior and posterior poles have the highest levels as desired (target in red). The dashed black lines represent the initial states, and the solid black lines depict the patterns during the last 100 time-steps.

3.2.2. The Gap Junctions and Cell Types Also Self-Organize into Patterns Even Though They Were Not Specifically Selected for That Purpose

The properties of the individual cells differentiate into characteristic states depending on the relative position of the cell, when initiated from homogeneous conditions (Figure 6). That is, the individual cells' properties also assume a pattern that reflects the system-level pattern, even though no specific associated targets were provided to them during training. This suggests that patterning at the lower levels (cell properties) may be necessary for patterning at the higher levels (cell states). Furthermore, it can be observed that the asymptotic patterns of the self-weights and cell types are almost bilaterally symmetrical. This means that the cells on the opposite sides of the center behave in symmetrical ways; oppositely signed activity levels tend to change in the opposite directions (as expected). The main reason why the network-level pattern of the asymptotic gap junction weights (Figure 6a) is not bilaterally symmetrical about the center is due to the requirement that the activity states of the anterior and posterior halves should be of opposite signs; thus, the anterior weights are more positive compared to the posterior side, resulting in relatively more positive change in the activity states of the anterior compared to the posterior (Figure 3).



(**a**) (**b**)

Figure 6. Cellular properties of the best-performing model. When initialized from homogeneous conditions and run for 4000 time-steps, the (**a**) intercellular gap-junction weights and the self-weights and (**b**) the cell types converge to characteristic shapes (black). Note that the model has eleven gap-junctions connecting the twelve cells in a chain, and every cell has a self-weight and a cell type.

The PCP-like structure of a single cell (Figure 7) is an important factor that contributes to both the formation of the network-level GJ weights (Figure 6a) and the activity pattern (Figure 5a). This PCP feature is reflected in the organization of the intrinsic controller, discovered by machine-learning, with positive weights of the anterior column and negative weights of the posterior column that, respectively, control the left and right GJs of the cell (Figure 7b). This symmetry-breaking phenomenon manifests in biology in a variety of equivalent ways on an ontogenetic timescale [73] and could be considered analogous to the reorganization of magnetic domains in ferromagnetic materials due to the application of external forces (Figure 7a).

3.2.3. The Model Successfully Regenerates and Rescales the Pattern despite Not Being Selected for Those Abilities

The model has not only learned to solve the patterning problem it was trained for, but also to regenerate and rescale patterns—abilities that were not rewarded for during the training. Specifically, when a fully developed pattern is partly reset, where just the state of a small portion in the middle portion is retained, the model successfully regenerates the rest of the pattern (Figure 8a). Likewise, when the model is started from almost double the number of cells under homogeneous conditions as before, the model develops a larger rescaled version of the original pattern (Figure 8b). These observations suggest that the model has learnt an abstract representation of the pattern independent of the details of its generator (the model). The information-processing strategy partly underlying this ability is that almost every intrinsic controller node in every cell contains information about the network-level pattern, as described below. We hypothesize that this system-level redundancy helps the network generate the pattern regardless of the network conditions.

3.2.4. The Model Generates the Same Qualitative Patterns Regardless of the Initial Network Conditions: Robustness

The model is surprisingly robust to a wider variety of initial conditions (Figure 9). That is, it not only regenerates and rescales patterns under homogeneous initial conditions, but it also canalizes random initial conditions (not seen during training) into the same qualitative patterns. In other words, the model exhibits characteristics of universal robustness, which could be attributed to a distributed information-representation mechanism, also hypothesized to be responsible for regeneration and rescaling, described in more detail below. Moreover, the model also shows a slight dependency on the initial conditions (since the simulations do not converge to the exact same pattern), analogous to biological embryos typically developing phenotypes with slight variations [74,75].

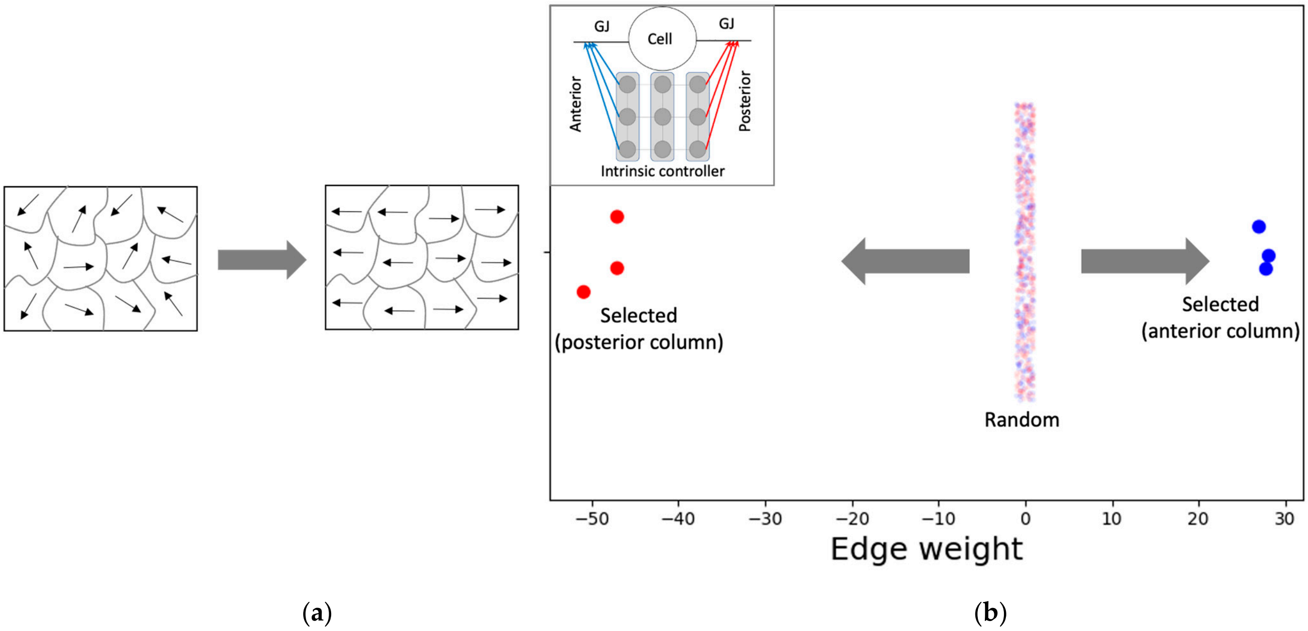


Figure 7. Single cells possess an intrinsic controller structure with characteristics resembling PCP. (**a**) A schematic illustrating the concept of PCP using the analogy of magnetic domains. Ferromagnetic materials contain domains within which the magnetic orientations are aligned (indicated by the arrows). The overall random pattern of orientations (left) could be modified by the application of external forces, such as magnetic fields, or temperature forcing it to assume non-random shapes (right). (**b**) Likewise, the application of a target gradient-like pattern (Figure 5) enables machine-learning to organize the initially random intrinsic controller weights into PCP-like patterns over phylogenetic timescales. Specifically, the three anterior controller weights (blue) that control the anterior GJ and the three posterior controller weights (red) that control the posterior GJ of a single cell were randomly initialized in the interval $[-1, 1]$ during the training. At the end of training, the anterior and posterior controller weights of the representative model culminated with categorically distinct values, the anterior set positive and the posterior negative, giving the cell a character of polarity. The inset shows a blow-up of a single cell together with its intrinsic controller (Figure 2).

3.3. Analysis of Intracellular Controller Activity Patterns

3.3.1. Internal Controller Activity Patterns Simultaneously Correlate with Cellular Properties and Network Activity Patterns

The states of the intrinsic controller nodes converge to patterns that simultaneously resemble the cell properties and the network activity patterns, depending on whether the states are normalized or not. Specifically, while the absolute intrinsic controller node states resemble the pattern of cellular properties (gap-junction weights, self-weights and cell types) (Figures 6 and 10a), their relative states resemble the network activity pattern (Figures 5a and 10b). This representation strategy makes sense since it is the same controller nodes that simultaneously influence the cell properties and are influenced by the cell state (Figure 2). Another interesting observation is that while the normalized activity state patterns of some of the intrinsic controller nodes resemble the original network activity pattern, others resemble its sign-flipped version (Figure 10b). One possible explanation for this partial inverse patterning activity is that it acts like a brake and helps the system balance the pattern, that is, it keeps the pattern from either exploding or flattening out (described in further detail in Section 3.4.2).

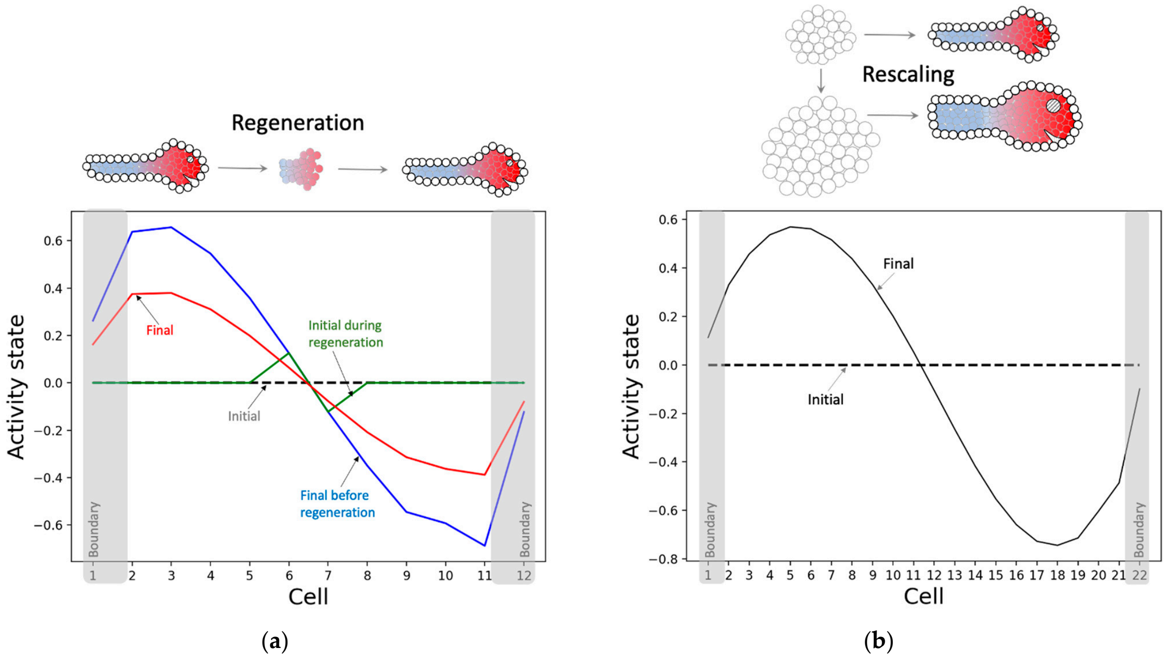


Figure 8. Regenerative and rescaling behaviors of the network activity. (**a**) Regeneration: the model is run for 4000 time-steps following homogeneous conditions, as before, leading to the blue pattern, then all states but that of the middle two cells are zeroed out (green) and run for another 4000 time-steps resulting in the final pattern (red). Even though the blue and red patterns do not exactly coincide they are qualitatively similar to each other. (**b**) Rescaling: the model is simulated in the same way as Figure 5a, except with 22 cells instead of the original 12 cells. With almost double the number of cells, the model takes about 3.5 times longer (14,000 time-steps) to settle, and moreover it converges (last 100 time-steps shown) to a smoother pattern compared to the 12-cell case.

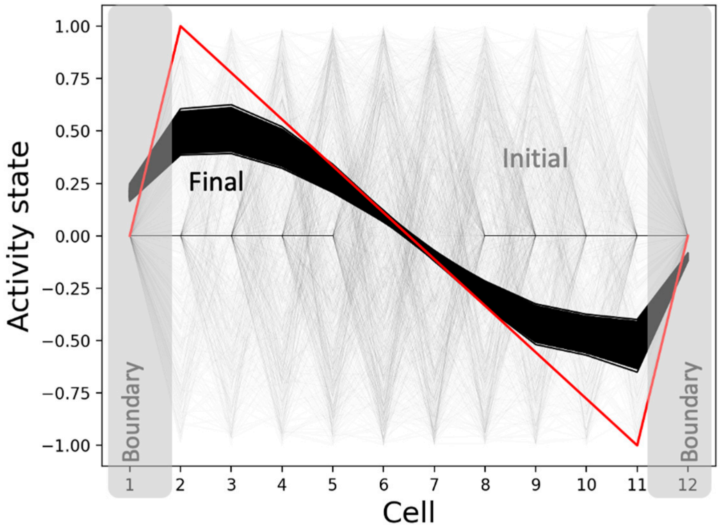


Figure 9. The pattern attractor-space of the activity state. The model converges to patterns (black) that are qualitatively similar to the target pattern (red) when started from a set of 1000 random initial conditions (grey). The initial conditions specifically involved a randomized initial number of 'active' cells whose activity states were drawn from the interval $[-1, 1]$ and boundary-marker states from the interval $[1, 2]$ with uniform probabilities. In the case of the 'non-active' cells, the activity states were set to 0 and the boundary-marker states were set to 2. The internal controller states were set to 0 in both cases.

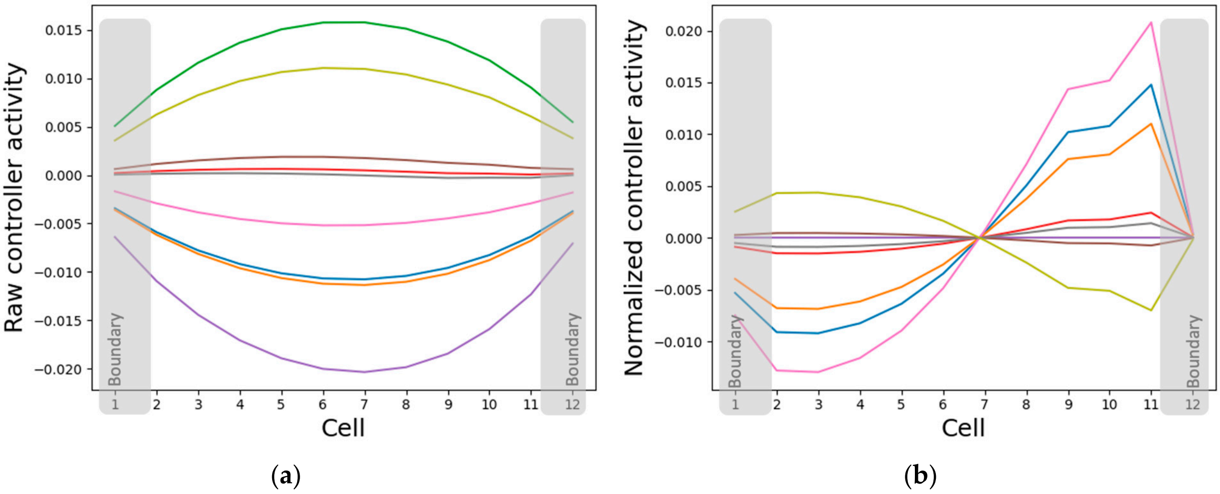


Figure 10. The intrinsic controller nodes' activities simultaneously resemble the cell-properties and the network-activity patterns. Each line in the plot represents the asymptotic activity of a particular controller node across the network. That is, each line represents (**a**) the vector $(r_{i,1}(\tau), \ldots, r_{i,n}(\tau))$ for a particular controller node $i \in \{1, \ldots, 9\}$ at $\tau = 4000$ and (**b**) its cell-normalized version $(\hat{r_{i1}}, \ldots, \hat{r_{in}})$ where $\hat{r_{ij}} = \frac{\left(r_{ij} - \min\limits_{1 \le j \le n} r_{ij}\right)}{\left(\max\limits_{1 \le j \le n} r_{ij} - \min\limits_{1 \le j \le n} r_{ij}\right)}$.

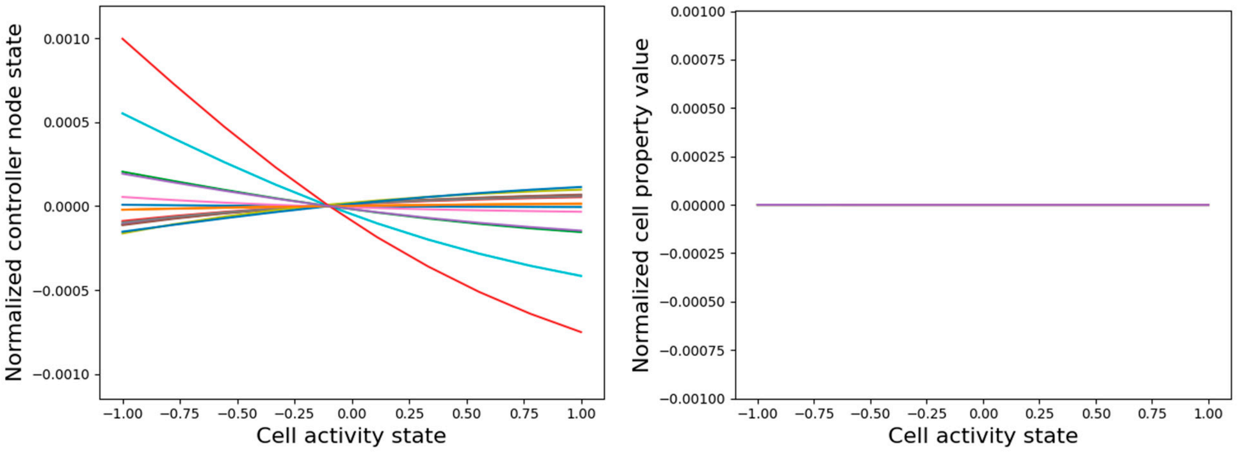
3.3.2. Isolated Cells Contain Relevant but Insufficient Information Required to Generate the Network-Level Patterns

A potential explanation for why the internal controller activity patterns correlate with the cellular activity patterns, as described above, is that the controllers are inherently fine-tuned to the cell's activity. That is, when a single cell is isolated from the network and the cell activity state is clamped, thus acting as the external input to the intrinsic controller, most controller node states tend to linearly correlate with the clamped input state (Figure 11a). However, the asymptotic GJs and cell type do not differentiate as a function of the input (Figure 11b). It is especially surprising that the GJ weights do not assume distinct values even though the anterior and posterior GJs are controlled by categorically distinct weights emanating from the intrinsic controller that give the cell a PCP-like character (Figure 7). These observations suggest that while the intrinsic controller is sensitive to single-cell activity, the cellular properties can only assume meaningful values in the collective context, a need that even the PCP-nature of the cell cannot mitigate. The learned sensitivity of the intrinsic controller is a sensible information-representation strategy, since the patterning activity states of a cell ultimately depend on the gap junction weights and the cell types that are in turn controlled by the internal signaling networks. On the other hand, it is evident that not all the information required for patterning is contained in single cells, as appropriately differentiated gap-junction weights and cell types are necessary for that purpose (Figure 6).

3.4. Analysis of Intercellular Causal Network Patterns

3.4.1. Every Cell in the Collective Contains the Full Causal Information about the Network-Level Patterns Explaining the Model's High Degree of Robustness

We measured the causal influence exerted by the initial state of every internal controller node of every cell over the asymptotic network activity. The resulting patterns of causal influence (Figure 12) closely resemble the phenotypic patterns themselves, suggesting that almost every intrinsic controller node of every cell contains information about the network activity pattern in the context of the network (isolated single cells contain only partial information, as described above). This observation points to a distributed and "universal" information-representation strategy that the model employs that could also explain its ability to generate the same patterns regardless of the initial network conditions as described above. In other words, the network is tightly integrated, and the global pattern information is accessible to every gene of every cell.



(**a**) (**b**)

Figure 11. Single cells contain relevant but insufficient dynamical information about the network-level pattern. When a single cell is isolated and its external input, namely the cell activity state, is clamped and simulated for 2000 time-steps (half the time required by the network to converge), then (**a**) the internal controller nodes converge to states that clearly discriminate between the various clamped inputs. However, (**b**) the cell properties, namely the cell type, the two gap-junction weights, the self-weights converge to the same values in the respective categories regardless of the clamped input. In both cases, the states were centered at their respective mean values.

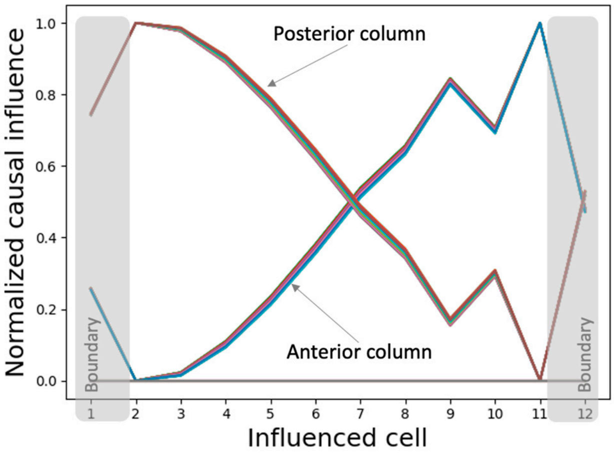


Figure 12. Individual nodes in the intrinsic controller network of every cell possess information about the network-level activity pattern that they control. Each line in the plot represents the normalized causal influence exerted by the initial state ($t = 0$) of a single internal controller node in a specific cell over the asymptotic ($\tau = 3500$) activity states of the (influenced) cells, that is, it is the normalized vector $\left(\frac{\partial s_1(\hat{3500})}{\partial r_{j,k}(0)}, \ldots, \frac{\partial s_n(\hat{3500})}{\partial r_{j,k}(0)}\right)$ for a specific controller node $j \in \{1, \ldots, 9\}$ in the influencing cell $k \in \{1, \ldots, n\}$ where, $\frac{\partial \hat{s_i(\tau)}}{\partial r_{j,k}(0)} = \frac{\frac{\partial s_i(\tau)}{\partial r_{j,k}(0)} - \min\limits_{1 \le i \le n} \frac{\partial s_i(\tau)}{\partial r_{j,k}(0)}}{\max\limits_{1 \le i \le n} \frac{\partial s_i(\tau)}{\partial r_{j,k}(0)} - \min\limits_{1 \le i \le n} \frac{\partial s_i(\tau)}{\partial r_{j,k}(0)}}$.

Furthermore, the various spatially segregated regions (columns) of the controllers exhibit symmetrically flipped causal influence patterns, reflecting their PCP-like organization at the network level (as labelled in Figure 12). Specifically, while the posterior column of the intrinsic controller display patterns resembling the network activity pattern, the anterior column exhibits a sign-flipped version of the same. These observations could again be partly attributed to the PCP-like organization at the level of the single cell (Figure 11). As noted above, this may also be a representation of the orientational symmetry of the axis.

3.4.2. The Network Dynamically Integrates into an Organization with Macro-Scale Modules Explaining the Overall Shape of the Functional Patterns

To make sense of the above results at the network level, we computed networks of causal influence between cells, where a connection from cell j to cell k represents a significant causal influence of the initial state of some internal controller node of j on the state of k at τ.

The resulting causal networks show features characteristic of increasing complexity, symmetry-breaking, long-range influence, and the emergence of macro-scale modules with increasing timescales (Figure 13). In particular, the modular organization of the causal network attractor suggests a high-level mechanism for the gradient-shape of the asymptotic network activity pattern. For instance, while the anterior half of the network influences itself with positive feedback loops it influences the posterior half with negative influence, partly explaining why the anterior half of the activity pattern state is positive-valued while the posterior half is negative-valued. The overall mixed-feedback organization explains why the whole pattern tends to balance itself (neither flattens out nor explodes). Likewise, the causal network attractor associated with the patterning of the boundary-marker reveals an organizer-like role played by the boundary cells in that they are the only cells that influence the rest of the network (Figure S6).

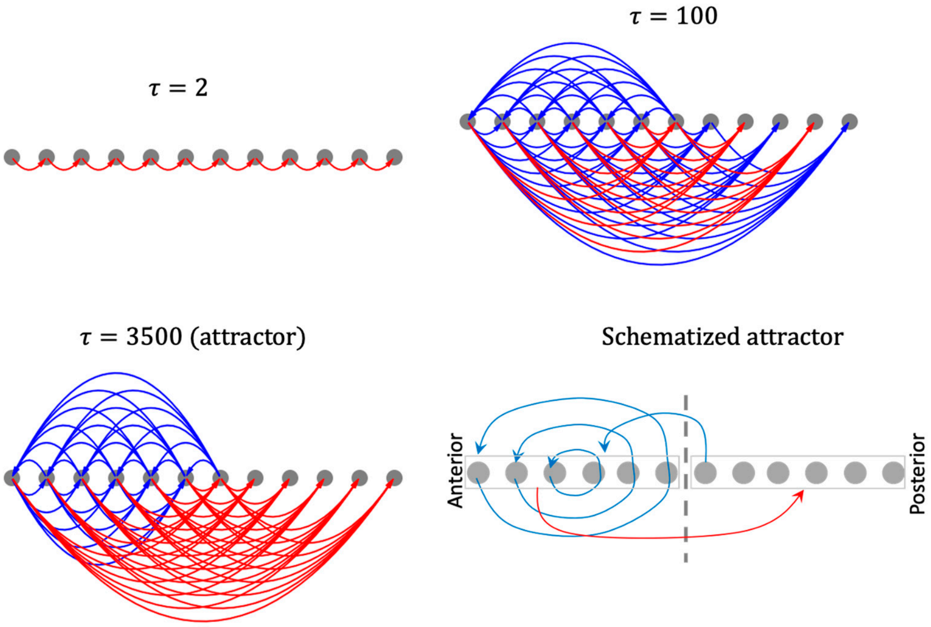


Figure 13. Causal network integration behind the network activity pattern developed under homogeneous initial conditions. An arrow from cell j to cell k represents the causal influence $\partial s_k(\tau)/\partial r_{i,j}(0)$ where $\exists i:\ \partial s_k(\tau)/\partial r_{i,j}(0)$ is a statistical outlier in the set $\left\{\frac{\partial s_k(\tau)}{\partial r_{1,j}(0)}, \ldots, \frac{\partial s_k(\tau)}{\partial r_{9,j}(0)}\right\}$. Blue links represent positive influence and red links represent negative influence. Multiple arrows originating from a cell may be associated with distinct intrinsic controller nodes of the originating cell.

This emergent modular organization cannot be explained by the structure of the original (symmetric) model itself, nor by that of the causal networks corresponding to lower timescales. In this way, the method of causal network integration partly helps close the gap between the structure and function of a complex dynamical model by focusing on the circuit-space rather than the conventional state-space.

3.4.3. Rescaling the Model Rescales the Causal Networks, Explaining Why the Phenotypic Patterns Rescale

The causal network integration approach offers further insights into the underlying mechanisms of patterning. It partly explains why rescaling the model (doubling the number of cells) results in the rescaling of the phenotypic patterns (Figure 8b)—the underlying causal network itself rescales (Figure 14). The macroscopic features of the rescaled causal network attractor (Figure 14b), for instance, preserve most of the modular structure of the original causal network (Figure 13), with the exception of the appearance of a couple of extra positive edges.

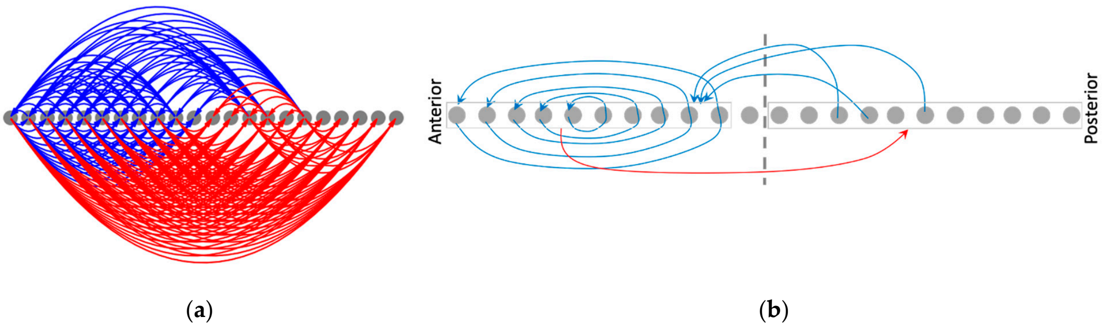


Figure 14. Rescaling the model (double the number of cells) rescales the corresponding causal network attractor underlying the network-activity pattern. The (**a**) causal network attractor and (**b**) its schematized version following rescaling of the model and simulating it with homogeneous initial conditions. The causal network attractor following regeneration is not shown, as it looks identical to the original (Figure 13).

3.4.4. The Overall Structure of the Mean Causal Network Explains the Model's Ability to Canalize Random Initial States into the Same Patterns

One of the emergent abilities of the model is to canalize random initial states (not seen during training) to similar pattern attractors (Figure 9). The reason is that the underlying causal networks corresponding to each of those random initial conditions themselves canalize into a mean attractor causal network whose overall features match those that corresponds to the homogenous case described above. For instance, the mean causal network attractor underlying network activity patterning (Figure 15) is characterized by the same anterior-positive and posterior-negative influences as the homogeneous case (Figure 13). The reason why the causal networks themselves canalize would involve investigations that are beyond the scope of this paper.

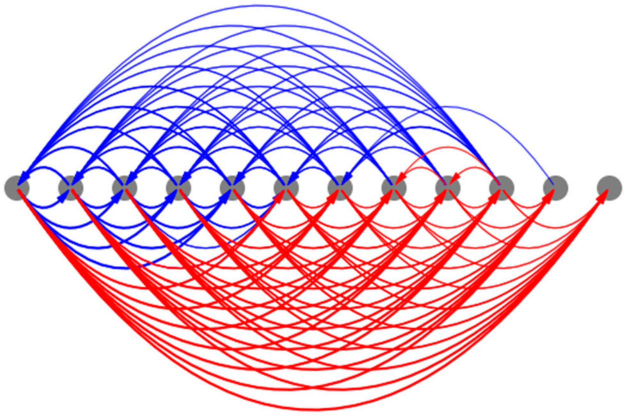


Figure 15. The mean causal network attractors associated with the network activity patterning. The thickness of the edges represents the frequency with which they appear in the set of attractors. The initial conditions that were used here are the same as those described in Figure 9.

4. Discussion

We have shown here that it is possible to train, using machine-learning, a recurrent self-organizing dynamical model incorporating biological priors to form gradient-like activity patterns from homogeneous conditions, that is, without the aid of externally supplied positional information or special initial conditions. We have also revealed the multi-timescale causal relationships among the components of the model, thereby describing the high-level mechanistic logic of pattern formation that the model employs.

One of the surprising findings of this work is the ability of the model to rescale the final pattern to an arbitrary number of cells (Figure 8b) despite not having been specifically trained to do so. This unexpected, emergent feature of this system mimics an interesting and important aspect of biology—plasticity. Numerous examples (for example, as reviewed in [13]) exist of robust, coherent organisms forming from the same genome despite drastic changes in the number, size, or type of cells [76–79]. The question of how certain types of search and encodings produce specifications of machinery with the ability to handle novel circumstances remains an open and important field of inquiry [80–82]. Our results reveal how physiological networks can embody a robust phenotypic patterning mechanism. We propose that this capability could be leveraged by evolution so that mutations resulting in an altered size of the organism need not require compensating mutations of the patterning mechanism. In other words, modularity potentiates evolution [83] by enabling a plasticity that allows organisms to maintain adaptive function (and thus fitness) while evolution explores changes in cell number.

From the perspective of morphogenesis, the causal network attractors described here could be conceived as a decoding of the "developmental program" encoded in the model's architecture and its learned parameters. In other words, these causal networks could be understood as forms of the developmental program itself. Moreover, the same underlying model could have multiple causal networks (different projections of a single developmental program), each responsible for a unique function, such as activity-patterning or boundary-marking, as we show here. In other words, starting from the 1st order "physiological" network we have uncovered 2nd order "physiocausal" networks. Thus, our work offers new perspectives and tools to achieve one of the major goals of developmental biology—to uncover the developmental programs that organisms use for morphogenesis [1].

From the perspective of computation, the causal networks could be viewed as high-level algorithms that the underlying biophysical machinery (model) employs for the purpose of axial pattern development. By "computation" we mean information-processing or transformation of information that serves a purpose (e.g., survival or adaptive function of an organism) [84]. One might ask—isn't the model an algorithm itself? It is what one might call a "low-level" algorithm. In a conventional algorithm (e.g., a computer program), there may be different paths that could be taken, via if-then conditions, for example. Exactly which path is taken depends on the inputs to the program. Moreover, a recurrent program that feeds the output back to itself could take different paths at different times depending on the dynamic inputs. All this suggests that even a conventional algorithm may not offer a complete explanation of the dynamic phenomenon it generates. This then raises the question: what is the ultimate high-level algorithm that describes how the given inputs are transformed into the final observed outputs in a non-recurrent and feed-forward manner? Our causal influence analysis offers a solution in this regard for our model—the causal network attractors are the high-level algorithms that offer a visual explanation for how the initial conditions are transformed into the final axial pattern. Through this analysis we also found that even though high-level algorithms, in principle, depend on initial conditions, they are all qualitatively similar to each other in our case (Figure 15). This makes sense in the context of development, as it ought to be robust enough to canalize multiple initial conditions to the same final pattern.

Viewed through the lens of the theory of computation, our model can be seen as an instance of autonomous sequential logic circuits—a type of finite state machine that does not involve external inputs. This is an appropriate class of models for developmental processes, as they are characteristically autonomous (with exceptions for environmentally triggered phenotypes, such as reviewed in [85]) and sequential at the large scale. The main limitation of this class of automata is that they are not capable of universal computation since they do not employ stacks or external tapes. At this point there is no indication that general-purpose computation is required of embryogenesis.

One of the striking features of the causal network attractors is long-distance influence: the relatively more significant asymptotic influence of the intrinsic controller of a cell on the activity state of another cell than itself, an emergent feature that is not baked in the original model (Figure 2). A biological analogue of this phenomenon would be the genes of one cell asymptotically controlling the features of another cell, that is, genetic control may not be local even though they may appear to generate only the features of the containing cell. If biological organisms indeed employ such an information-processing principle, then it would have therapeutic implications such as non-local genetic intervention. For example, might a more effective gene therapy for cancer require the hacking of the genes of neighboring healthy cells rather than themselves? Another striking feature of the causal network attractors is their modularity; cells organize into modules that tend to contain causal influences amongst themselves. Modularity is not a new concept in biology, although conventional views have focused on the overt model structure [86], and only recently has the focus widened to encompass its dynamical aspects [87–89]. The causal network integration approach offers a novel perspective on dynamical modularity through temporally integrated models. Such higher-order structures may lie waiting to be discovered in a variety of published biological regulatory models, containing potential high-level insights. For instance, could it be possible that the capacity of GRNs for associative memory [90] is due to a high-level causal network that is equivalent to a characteristic minimal network that is necessary and sufficient for the implementation of the memory? Overall, the tools developed here provide a new lens through which to view emergent phenomena.

Our work also offers new tools for solving the problem of top-down control in biology [13,91], where one of the open challenges is to systematically edit a complex regulatory system so that it generates a desired outcome. One way the causal network integration approach may mitigate this challenge is by offering a way to close the gap between the structure and function of a complex system by focusing on the circuit-space of the system. By offering a circuit-centric explanation of the function of a system at the top-most (asymptotic) timescale, this method thereby offers a systematic way to modify the underlying model by working the changes back down to the smallest timescale. Even though we have not worked out the details of how it could be implemented, we suspect that it would involve inferring a higher-order network model that dictates how the causal networks themselves change over time. A successful solution to the problem of top-down control would have a wide impact on biology via prescriptions for systematic interventions into biochemical networks that underlie disease. It would also impact the field of machine-learning by way of novel mechanisms of systematic supervised learning that could leverage the information contained in the macroscale structure of the causal networks.

Our work shows that the method of causal network integration has the potential to generate multi-timescale insights into how information is organized in the network. This approach of the analysis of a dynamical system focuses on the circuit-space (the space of circuits, as opposed to the space of states, induced by the model), whereas conventional approaches tend to focus on the state-space of the system. By casting the emergent dynamics at multiple scales in the circuit space, this approach brings us a step closer to closing the gap between the structure and the function of a complex dynamical system. In this regard, our method also contributes to the theory of complex systems by complementing and potentially generalizing existing approaches to characterizing canalization [88], control [92], collectivity [70,93], coarse-graining [94–96] and criticality in complex nonlinear dynamical systems [97,98].

All the code developed for this project can be found at: https://gitlab.com/smanicka/MinimalDevelopmentalComputation (accessed on: 4 January 2022).

Supplementary Materials: The following supporting information can be downloaded at: https://www.mdpi.com/article/10.3390/e24010107/s1: Figure S1: Behavior of the best-performing model is representative of the ensemble of the top-performing models; Figure S2: Regenerative and rescaling behaviors of the boundary-marker; Figure S3: The pattern attractor-space of the boundary-marker level; Figure S4: The boundary controller nodes' activities simultaneously resemble the boundary-marker and the network-activity patterns; Figure S5: Individual nodes in the boundary controller network of every cell possess information about the network-level boundary-marker pattern that they control; Figure S6: Causal network integration behind the boundary-marker pattern developed under homogeneous initial conditions; Figure S7: Rescaling the model (double the number of cells) rescales the corresponding causal network attractor underlying the boundary-marker pattern; Figure S8: The mean causal network attractors associated with the boundary-marker patterning.

Author Contributions: Conceptualization, S.M. and M.L.; methodology, S.M. and M.L.; software, S.M.; validation, S.M. and M.L.; formal analysis, S.M.; investigation, S.M. and M.L.; resources, M.L.; data curation, S.M.; writing—original draft preparation, S.M.; writing—review and editing, S.M. and M.L.; visualization, S.M. and M.L.; supervision, M.L.; project administration, M.L.; funding acquisition, M.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research was made possible by the NSF (grant 2020247), the Guy Foundation Family trust (103733-00001), and Grant 62212 from the John Templeton Foundation (the opinions expressed in this publication are those of the author(s) and do not necessarily reflect the views of the John Templeton Foundation).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data sharing not applicable.

Acknowledgments: We thank Vaibhav Pai and Patrick McMillen for providing helpful comments on a draft of the manuscript.

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results.

References

1. Slack, J.M.W. *From Egg to Embryo: Regional Specification in Early Development*; Cambridge University Press: Cambridge UK, 1991.
2. Child, C.M. *Patterns and Problems of Development*; The University of Chicago Press: Chicago, IL, USA, 1941.
3. Siregar, P.; Julen, N.; Hufnagl, P.; Mutter, G. A general framework dedicated to computational morphogenesis Part I—Constitutive equations. *Biosystems* **2018**, *173*, 298–313. [CrossRef] [PubMed]
4. Siregar, P.; Julen, N.; Hufnagl, P.; Mutter, G. A general framework dedicated to computational morphogenesis Part II—Knowledge representation and architecture. *Biosystems* **2018**, *173*, 314–334. [CrossRef]
5. Morelli, L.G.; Uriu, K.; Ares, S.; Oates, A.C. Computational approaches to developmental patterning. *Science* **2012**, *336*, 187–191. [CrossRef] [PubMed]
6. Hogeweg, P. Computing an organism: On the interface between informatic and dynamic processes. *BioSystems* **2002**, *64*, 97–109. [CrossRef]
7. Wolpert, L.; Lewis, J.H. Towards a theory of development. *Fed. Proc.* **1975**, *34*, 14–20.
8. Slack, J.M. A serial threshold theory of regeneration. *J. Theor. Biol.* **1980**, *82*, 105–140. [CrossRef]
9. Slack, J.M.W. Serial Thresholds and Homeotic Variation. In *Kinetic Logic: A Boolean Approach to the Analysis of Complex Regulatory Systems*; Thomas, R., Ed.; Springer: Berlin/Heidelberg, Germany, 1979; Volume 29.
10. Allen, M.; Friston, K.J. From cognitivism to autopoiesis: Towards a computational framework for the embodied mind. *Synthese* **2018**, *195*, 2459–2482. [CrossRef]
11. Keijzer, F.A. Evolutionary convergence and biologically embodied cognition. *Interface Focus* **2017**, *7*, 20160123. [CrossRef] [PubMed]
12. Ziemke, T. The body of knowledge: On the role of the living body in grounding embodied cognition. *Biosystems* **2016**, *148*, 4–11. [CrossRef]
13. Pezzulo, G.; Levin, M. Top-down models in biology: Explanation and control of complex living systems above the molecular level. *J. R. Soc. Interface* **2016**, *13*, 20160555. [CrossRef]
14. Pinet, K.; McLaughlin, K.A. Mechanisms of physiological tissue remodeling in animals: Manipulating tissue, organ, and organism morphology. *Dev. Biol.* **2019**, *451*, 134–145. [CrossRef]

15. Harris, A.K. The need for a concept of shape homeostasis. *Biosystems* **2018**, *173*, 65–72. [CrossRef]
16. Mustard, J.; Levin, M. Bioelectrical Mechanisms for Programming Growth and Form: Taming Physiological Networks for Soft Body Robotics. *Soft Robot.* **2014**, *1*, 169–191. [CrossRef]
17. Sharpe, J. Computer modeling in developmental biology: Growing today, essential tomorrow. *Development* **2017**, *144*, 4214–4225. [CrossRef] [PubMed]
18. Pastor-Escuredo, D.; del Álamo, J.C. How Computation Is Helping Unravel the Dynamics of Morphogenesis. *Front. Phys.* **2020**, *8*, 31. [CrossRef]
19. Chara, O.; Tanaka, E.M.; Brusch, L. Mathematical modeling of regenerative processes. *Curr. Top. Dev. Biol.* **2014**, *108*, 283–317.
20. Meinhardt, H. Pattern formation in biology: A comparison of models and experiments. *Rep. Prog. Phys.* **1992**, *55*, 797–849. [CrossRef]
21. Umulis, D.M.; Othmer, H.G. The role of mathematical models in understanding pattern formation in developmental biology. *Bull. Math. Biol.* **2015**, *77*, 817–845. [CrossRef]
22. Gierer, A.; Meinhardt, H. A theory of biological pattern formation. *Kybernetik* **1972**, *12*, 30–39. [CrossRef]
23. Briscoe, J. Understanding Pattern Formation in Embryos: Experiment, Theory, and Simulation. *J. Comput. Biol.* **2019**, *26*, 696–702. [CrossRef]
24. Maini, P.K. Using mathematical models to help understand biological pattern formation. *Comptes Rendus Biol.* **2004**, *327*, 225–234. [CrossRef]
25. Wolpert, L. Positional information and the spatial pattern of cellular differentiation. *J. Theor. Biol.* **1969**, *25*, 1–47. [CrossRef]
26. Wolpert, L. *The French Flag Problem: A Contribution to the Discussion on Pattern Deve Lopment and Regulation, towards a Theoretical Biology*; Waddington, C.H., Ed.; Edinburgh University Press: Edinburgh, UK, 1968; Volume 1.
27. Sharpe, J. Wolpert's French Flag: What's the problem? *Development* **2019**, *146*, dev185967. [CrossRef]
28. Bryant, P.J.; Bryant, S.V.; French, V. Biological regeneration and pattern formation. *Sci. Am.* **1977**, *237*, 66–81. [CrossRef] [PubMed]
29. Agata, K.; Saito, Y.; Nakajima, E. Unifying principles of regeneration I: Epimorphosis versus morphallaxis. *Dev. Growth Differ.* **2007**, *49*, 73–78. [CrossRef] [PubMed]
30. Martinez Arias, A.; Steventon, B. On the nature and function of organizers. *Development* **2018**, *145*, dev159525. [CrossRef] [PubMed]
31. Cooke, J.; Zeeman, E.C. A Clock and Wavefront Model for Control of the Number of Repeated Structures during Animal Morphogenesis. *J. Theor. Biol.* **1976**, *58*, 455–476. [CrossRef]
32. Turing, A.M. The chemical basis of morphogenesis. *Bull. Math. Biol.* **1990**, *52*, 153–197. [CrossRef]
33. Azuaje, F. Computational discrete models of tissue growth and regeneration. *Brief. Bioinform.* **2011**, *12*, 64–77. [CrossRef]
34. Minh-Thai, T.N.; Aryal, J.; Samarasinghe, S.; Levin, M. A Computational Framework for Autonomous Self-repair Systems. In *AI 2018: Advances in Artificial Intelligence*; Springer: New York, NY, USA, 2018; pp. 153–159.
35. De, A.; Chakravarthy, V.S.; Levin, M. A computational model of planarian regeneration. *Int. J. Parallel Emergent Distrib. Syst.* **2016**, *32*, 331–347. [CrossRef]
36. Tosenberger, A.; Bessonov, N.; Levin, M.; Reinberg, N.; Volpert, V.; Morozova, N. A conceptual model of morphogenesis and regeneration. *Acta Biotheor.* **2015**, *63*, 283–294. [CrossRef] [PubMed]
37. Bessonov, N.; Levin, M.; Morozova, N.; Reinberg, N.; Tosenberger, A.; Volpert, V. On a Model of Pattern Regeneration Based on Cell Memory. *PLoS ONE* **2015**, *10*, e0118091. [CrossRef] [PubMed]
38. Bessonov, N.; Levin, M.; Morozova, N.; Reinberg, N.; Tosenberger, A.; Volpert, V. Target morphology and cell memory: A model of regenerative pattern formation. *Neural Regen. Res.* **2015**, *10*, 1901–1905. [PubMed]
39. Lobo, D.; Solano, M.; Bubenik, G.A.; Levin, M. A linear-encoding model explains the variability of the target morphology in regeneration. *J. R. Soc. Interface* **2014**, *11*, 20130918. [CrossRef]
40. Harrington, K.I.S. A circuit basis for morphogenesis. *Theor. Comput. Sci.* **2016**, *633*, 28–36. [CrossRef]
41. Voosen, P. The AI Detectives. *Science* **2017**, *357*, 22–27. [CrossRef]
42. Beer, R.D.; Williams, P.L. Information processing and dynamics in minimally cognitive agents. *Cogn. Sci.* **2015**, *39*, 1–38. [CrossRef]
43. Tkačik, G.; Bialek, W. Information processing in living systems. *Annu. Rev. Condens. Matter Phys.* **2016**, *7*, 89–117. [CrossRef]
44. Gatenby, R.A.; Frieden, B.R. Information theory in living systems, methods, applications, and challenges. *Bull. Math. Biol.* **2007**, *69*, 635–657. [CrossRef]
45. Mannino, M.; Bressler, S.L. Foundational perspectives on causality in large-scale brain networks. *Phys. Life Rev.* **2015**, *15*, 107–123. [CrossRef]
46. Chicharro, D.; Ledberg, A. When two become one: The limits of causality analysis of brain dynamics. *PLoS ONE* **2012**, *7*, e32466. [CrossRef]
47. Friston, K.J.; Harrison, L.; Penny, W. Dynamic causal modelling. *Neuroimage* **2003**, *19*, 1273–1302. [CrossRef]
48. Linardatos, P.; Papastefanopoulos, V.; Kotsiantis, S. Explainable ai: A review of machine learning interpretability methods. *Entropy* **2021**, *23*, 18. [CrossRef] [PubMed]
49. Simonyan, K.; Vedaldi, A.; Zisserman, A. Deep inside convolutional networks: Visualising image classification models and saliency maps. *arXiv* **2013**, arXiv:1312.6034.
50. Fong, R.; Patrick, M.; Vedaldi, A. Understanding Deep Networks Via Extremal Perturbations and Smooth Masks. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Korea, 27 October–2 November 2019; pp. 2950–2958.

51. Hernández, A.; Amigó, J.M. Differentiable programming and its applications to dynamical systems. *arXiv* **2019**, arXiv:1912.08168.
52. Durston, A.J.; Zhu, K. A time space translation hypothesis for vertebrate axial patterning. *Semin. Cell Dev. Biol.* **2015**, *42*, 86–93. [CrossRef]
53. Durston, A.J.; Jansen, H.J.; Wacker, S.A. Review: Time-space translation regulates trunk axial patterning in the early vertebrate embryo. *Genomics* **2010**, *95*, 250–255. [CrossRef]
54. Houston, D.W. Vertebrate Axial Patterning: From Egg to Asymmetry. *Adv. Exp. Med. Biol.* **2017**, *953*, 209–306.
55. Delgado, I.; Torres, M. Coordination of limb development by crosstalk among axial patterning pathways. *Dev. Biol.* **2017**, *429*, 382–386. [CrossRef]
56. Bode, H.R. Axial patterning in hydra. *Cold Spring Harb. Perspect. Biol.* **2009**, *1*, a000463. [CrossRef]
57. Whitfield, T.T.; Hammond, K.L. Axial patterning in the developing vertebrate inner ear. *Int. J. Dev. Biol.* **2007**, *51*, 507–520. [CrossRef]
58. Wellik, D.M. Hox patterning of the vertebrate axial skeleton. *Dev. Dyn.* **2007**, *236*, 2454–2463. [CrossRef]
59. Bowman, J.L. Axial patterning in leaves and other lateral organs. *Curr. Opin. Genet. Dev.* **2000**, *10*, 399–404. [CrossRef]
60. Stocum, D.L. A conceptual framework for analyzing axial patterning in regenerating urodele limbs. *Int. J. Dev. Biol.* **1996**, *40*, 773–783. [PubMed]
61. Mathews, J.; Levin, M. Gap junctional signaling in pattern regulation: Physiological network connectivity instructs growth and form. *Dev. Neurobiol.* **2017**, *77*, 643–673. [CrossRef]
62. Pandit, N.K. *Introduction to the Pharmaceutical Sciences*; Lippincott Williams & Wilkins: Philadelphia, PA, USA, 2007.
63. Palacios-Prado, N.; Bukauskas, F.F. Heterotypic gap junction channels as voltage-sensitive valves for intercellular signaling. *Proc. Natl. Acad. Sci. USA* **2009**, *106*, 14855–14860. [CrossRef]
64. Roberts, K.; Alberts, B.; Johnson, A.; Walter, P.; Hunt, T. *Molecular Biology of the Cell*; Garland Science: New York, NY, USA, 2002.
65. Bischof, J.; Day, M.E.; Miller, K.A.; LaPalme, J.V.; Levin, M. Nervous system and tissue polarity dynamically adapt to new morphologies in planaria. *Dev. Biol.* **2020**, *467*, 51–65. [CrossRef]
66. Pietak, A.; Bischof, J.; LaPalme, J.; Morokuma, J.; Levin, M. Neural control of body-plan axis in regenerating planaria. *PLoS Comput. Biol.* **2019**, *15*, e1006904. [CrossRef] [PubMed]
67. Levin, M. Bioelectric signaling: Reprogrammable circuits underlying embryogenesis, regeneration, and cancer. *Cell* **2021**, *184*, 1971–1989. [CrossRef] [PubMed]
68. Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.; Antiga, L. Pytorch: An imperative style, high-performance deep learning library. *Adv. Neural Inf. Processing Syst.* **2019**, *32*, 8026–8037.
69. Liu, Y.-Y.; Slotine, J.-J.; Barabási, A.-L. Controllability of complex networks. *Nature* **2011**, *473*, 167–173. [CrossRef] [PubMed]
70. Brush, E.R.; Krakauer, D.C.; Flack, J.C. A family of algorithms for computing consensus about node state from network data. *PLoS Comput. Biol.* **2013**, *9*, e1003109. [CrossRef] [PubMed]
71. Manicka, S.V.S. *The Role of Canalization in the Spreading of Perturbations in Boolean Networks*; Indiana University: Bloomington, IN, USA, 2017.
72. Lizier, J.T.; Prokopenko, M.; Zomaya, A.Y. A framework for the local information dynamics of distributed computation in complex systems. In *Guided Self-Organization: Inception*; Springer: New York, NY, USA, 2014; pp. 115–158.
73. Wang, Y.; Nathans, J. Tissue/planar cell polarity in vertebrates: New insights and new questions. *Development* **2007**, *134*, 647–658. [CrossRef] [PubMed]
74. Aranda-Anzaldo, A.; Dent, M.A. Developmental noise, ageing and cancer. *Mech. Ageing Dev.* **2003**, *124*, 711–720. [CrossRef]
75. Woods, H.A. Mosaic physiology from developmental noise: Within-organism physiological diversity as an alternative to phenotypic plasticity and phenotypic flexibility. *J. Exp. Biol.* **2014**, *217 Pt 1*, 35–45. [CrossRef]
76. Cooke, J. Scale of body pattern adjusts to available cell number in amphibian embryos. *Nature* **1981**, *290*, 775–778. [CrossRef]
77. Fankhauser, G. Maintenance of normal structure in heteroploid salamander larvae, through compensation of changes in cell size by adjustment of cell number and cell shape. *J. Exp. Zool.* **1945**, *100*, 445–455. [CrossRef]
78. Kriegman, S.; Blackiston, D.; Levin, M.; Bongard, J. Kinematic self-replication in reconfigurable organisms. *Proc. Natl. Acad. Sci. USA* **2021**, *118*, e2112672118. [CrossRef]
79. Blackiston, D.; Lederer, E.; Kriegman, S.; Garnier, S.; Bongard, J.; Levin, M. A cellular platform for the development of synthetic living machines. *Sci. Robot.* **2021**, *6*, eabf1571. [CrossRef]
80. Watson, R.A.; Wagner, G.P.; Pavlicev, M.; Weinreich, D.M.; Mills, R. The evolution of phenotypic correlations and "developmental memory". *Evolution* **2014**, *68*, 1124–1138. [CrossRef]
81. Parter, M.; Kashtan, N.; Alon, U. Facilitated variation: How evolution learns from past environments to generalize to new environments. *PLoS Comput. Biol.* **2008**, *4*, e1000206. [CrossRef]
82. Gerhart, J.; Kirschner, M. The theory of facilitated variation. *Proc. Natl. Acad. Sci. USA* **2007**, *104* (Suppl. S1), 8582–8589. [CrossRef] [PubMed]
83. Melo, D.; Porto, A.; Cheverud, J.M.; Marroig, G. Modularity: Genes, development, and evolution. *Annu. Rev. Ecol. Evol. Syst.* **2016**, *47*, 463–486. [CrossRef]
84. Mitchell, M. Ubiquity symposium: Biological computation. *Ubiquity* **2011**, *2011*, 1–7. [CrossRef]
85. Tung, A.; Levin, M. Extra-genomic instructive influences in morphogenesis: A review of external signals that regulate growth and form. *Dev. Biol.* **2020**, *461*, 1–12. [CrossRef] [PubMed]

86. Hartwell, L.H.; Hopfield, J.J.; Leibler, S.; Murray, A.W. From molecular to modular cell biology. *Nature* **1999**, *402*, C47–C52. [CrossRef]
87. Kolchinsky, A.; Rocha, L.M. Prediction and modularity in dynamical systems. *arXiv* **2011**, arXiv:1106.3703.
88. Marques-Pita, M.; Rocha, L.M. Canalization and control in automata networks: Body segmentation in Drosophila melanogaster. *PLoS ONE* **2013**, *8*, e55946. [CrossRef]
89. Jaeger, J.; Monk, N. Dynamical modules in metabolism, cell and developmental biology. *Interface Focus* **2021**, *11*, 20210011. [CrossRef]
90. Biswas, S.; Manicka, S.; Hoel, E.; Levin, M. Gene regulatory networks exhibit several kinds of memory: Quantification of memory in biological and random transcriptional networks. *Iscience* **2021**, *24*, 102131. [CrossRef] [PubMed]
91. Walker, S.I. Top-down causation and the rise of information in the emergence of life. *Information* **2014**, *5*, 424–439. [CrossRef]
92. Gates, A.J.; Correia, R.B.; Wang, X.; Rocha, L.M. The effective graph reveals redundancy, canalization, and control pathways in biochemical regulation and signaling. *Proc. Natl. Acad. Sci. USA* **2021**, *118*, e2022598118. [CrossRef]
93. Daniels, B.C.; Ellison, C.J.; Krakauer, D.C.; Flack, J.C. Quantifying collectivity. *Curr. Opin. Neurobiol.* **2016**, *37*, 106–113. [CrossRef]
94. Flack, J.C. Coarse-graining as a downward causation mechanism. *Philos. Trans. A Math. Phys. Eng. Sci.* **2017**, *375*, 20160338. [CrossRef] [PubMed]
95. Hoel, E. When the Map Is Better Than the Territory. *Entropy* **2017**, *19*, 188. [CrossRef]
96. Klein, B.; Hoel, E. The Emergence of Informative Higher Scales in Complex Networks. *Complexity* **2020**, *2020*, 1–12. [CrossRef]
97. Kim, H.; Sayama, H. The role of criticality of gene regulatory networks in morphogenesis. *IEEE Trans. Cogn. Dev. Syst.* **2018**, *12*, 390–400. [CrossRef]
98. Daniels, B.C.; Kim, H.; Moore, D.; Zhou, S.; Smith, H.B.; Karas, B.; Kauffman, S.A.; Walker, S.I. Criticality Distinguishes the Ensemble of Biological Regulatory Networks. *Phys. Rev. Lett.* **2018**, *121*, 138102. [CrossRef] [PubMed]