

Quantifying Different Modeling Frameworks Using Topological Data Analysis: A Case Study with Zebrafish Patterns*

Electa Cleveland[†], Angela Zhu[†], Björn Sandstede[†], and Alexandria Volkening[‡]

Abstract. Mathematical models come in many forms across biological applications. In the case of complex, spatial dynamics and pattern formation, stochastic models also face two main challenges: pattern data are largely qualitative, and model realizations may vary significantly. Together these issues make it difficult to relate models and empirical data—or even models and models—limiting how different approaches can be combined to offer new insights into biology. These challenges also raise mathematical questions about how models are related, since alternative approaches to the same problem—e.g., Cellular Potts models; off-lattice, agent-based models; on-lattice, cellular automaton models; and continuum approaches—treat uncertainty and implement cell behavior in different ways. To help open the door to future work on questions like these, here we adapt methods from topological data analysis and computational geometry to quantitatively relate two different models of the same biological process in a fair, comparable way. To center our work and illustrate concrete challenges, we focus on the example of zebrafish-skin pattern formation, and we relate patterns that arise from agent-based and cellular automaton models.

Key words. agent-based model, cellular automaton, topological data analysis, alpha-shapes, variability, zebrafish, pattern formation

MSC codes. 92B05, 92C15, 70G60, 55N31

DOI. 10.1137/22M1543082

1. Introduction. Whether focused on vegetation patterns [45], schooling and swarming [6, 57], collective cell dynamics [9, 14, 43, 77, 99], or another complex system, researchers use a wide range of mathematical models to study pattern formation. These approaches include stochastic individual-based models, which may operate on or off lattice, and macroscopic models in the form of partial differential equations. Understanding how the predictions that arise across these frameworks are related allows us to assess the impact of modeling assumptions. Importantly, the same biological mechanism can be implemented in different ways in different models, raising questions about how dependent a model's inferences about a given mechanism are on its implementation. Quantitative model comparison helps address this by bringing models together, increasing biological insight by providing more than one perspective. However, relating different models of the same biological system is not straightforward, partic-

*Received by the editors December 23, 2022; accepted for publication (in revised form) by A. Layton July 16, 2023; published electronically November 29, 2023.

<https://doi.org/10.1137/22M1543082>

Funding: The work of the first and second authors was supported in part by the NSF through grant DMS-1714429 and by Brown University through Karen T. Romer Undergraduate Teaching and Research Awards. The work of the third author was partially supported by the NSF through grants DMS-2038039 and DMS-2106566.

[†]Division of Applied Mathematics, Brown University, Providence, RI 02912 USA (electa_cleveland@brown.edu, angela_zhu1@brown.edu, bjorn_sandstede@brown.edu).

[‡]Department of Mathematics, Purdue University, West Lafayette, IN 47907 USA (avolkening@purdue.edu).

ularly when considering spatial dynamics and detailed, stochastic models, which often rely on qualitative observation to judge model output. Motivated by this challenge, we show how to adapt methods from topological data analysis and computational geometry to quantitatively describe two models of the same biological system in a comparable way. For concreteness, we center our study on zebrafish-skin patterns, and we apply our methods to two stochastic, individual-based models [80, 102].

As we show in Figure 1, zebrafish (*Danio rerio*) are small fish known for their black stripes and gold interstripes [55, 82, 99]. These stripe patterns are made up of brightly colored pigment cells and emerge from pigment-cell interactions and the tissue environment [64, 82, 84, 108]. The diversity of patterns that mutant fish sport when cell interactions are altered makes zebrafish skin an attractive place to study pattern formation mathematically [58, 99]. Toward this end, zebrafish patterns have been investigated using many of the modeling approaches that are also used to study biological self-organization in other applications. Microscopic off-lattice [98, 100, 101, 102, 106] and on-lattice [13, 59, 73, 80] models have considered the behavior of individual cells. On the macroscopic side, continuum models [10, 13, 38, 59, 75, 81, 105, 108], including reaction-diffusion equations and integro-differential equations, have been developed to track cell densities. Other approaches [67] include reducing complexity by only describing the evolution of the stripe–interstripe interface.

For zebrafish skin and other biological patterns, continuum models have the benefit of being analytically tractable. They also provide a broad perspective on the overarching features (such as density-dependent motion [15, 63] or short-range activation and long-range inhibition [42, 96]) that may be at work. On the other hand, agent-based and cellular automaton models,¹ not analytically tractable using traditional techniques, are often closer to the underlying

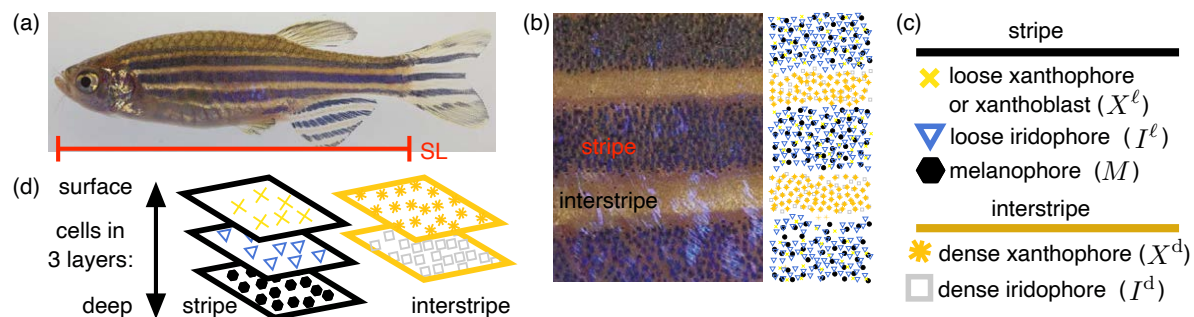


Figure 1. Biological background. (a) Zebrafish (*Danio rerio*) are a model organism for studying vertebrate development and skin-pattern formation [55, 88, 99]. (b) Their stripes and interstripes form due to the interactions of pigment cells [75]. (c) Stripes feature X^l , I^l , and M cells, and interstripes consist of X^d and I^d cells. (d) These cells are organized in three layers, with xanthophores at the surface level [51, 66, 70, 84]. Images (a) and (b) are adapted from Fadeev et al. [33] and licensed under CC-BY 4.0 (<https://creativecommons.org/licenses/by/4.0/>); we added the SL bar, text, and schematic in (a) and (b); images (c) and (d) are adapted from Volkening et al. [102] under CC-BY 4.0.

¹Researchers use the terms “agent-based,” “individual-based,” and “cellular automaton” to describe a rich diversity of models. In this manuscript, “agent-based” means a model that works at the scale of agents—e.g., cells—interacting in continuous space. We distinguish this from what we call “cellular automaton” models, which also focus on the dynamics of agents, but in which space is discrete; see Figure 2.

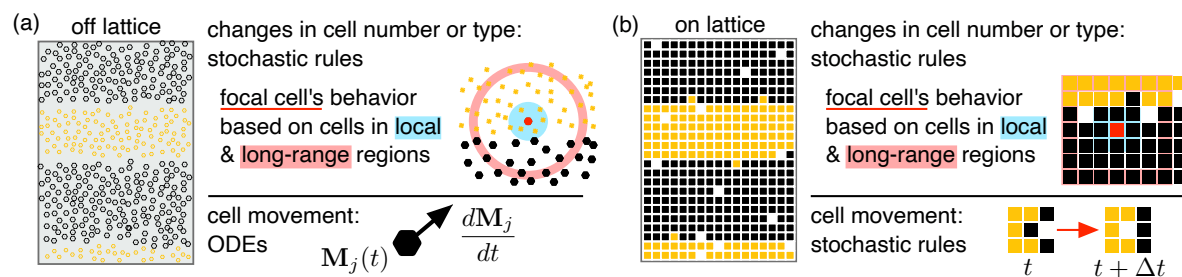


Figure 2. Motivation. (a) Prior modeling approaches to pattern formation in zebrafish include stochastic agent-based (off-lattice microscopic) models and cellular automaton (on-lattice microscopic) models [99]. Agent-based models [98, 100, 101, 102] treat cells as particles that interact through ODEs for cell movement and stochastic, discrete-time rules for differentiation, division, and death. (b) Cellular automaton models [13, 59, 80] consider space as discrete, and all cell interactions are stochastic rules. Both of these approaches are not analytically tractable using traditional techniques, and they often involve qualitative observations to judge model output. This makes it difficult to relate different models, and this challenge motivates our study. Images (a) and (b) are adapted from Volkening [99] with permission from Elsevier. Copyright (2020) Elsevier Ltd.

biology and can make detailed predictions about cell interactions. Uncovering how different types of models are related is a rich mathematical problem. Moreover, because it breaks down silos and allows us to bring the predictions of different studies together, it has important biological applications. To this end, much work has focused on relating macroscopic and microscopic approaches. In the case of zebrafish patterns, for example, Bullara et al. [13], Konow et al. [59], and Volkening et al. [98, 101] developed continuum models for cell density alongside microscopic models. While open questions certainly remain in relating macroscopic and microscopic models, much less work has focused on understanding the relationship among different microscopic models, such as on-lattice and off-lattice approaches. We highlight the studies of Osborne et al. [77] and Plank et al. [89] on cell behavior as examples.

The increased level of biological detail in individual-based models comes at the cost of more parameters and model rules. The computational complexity of microscopic models—whether they are, for example, agent-based, cellular automaton, Cellular Potts [44, 49], or vertex-based models—also leads to more choices and hidden parameters. (See Buttenschön et al. [14] for a recent review of computational models of cell behavior.) This raises questions about how choices of computational implementation affect model predictions [60, 77]. However, the stochastic, complex nature of individual-based models makes it challenging to quantitatively describe—let alone robustly compare and relate—model output. On top of this, qualitative observation of a few example simulations may not capture the breadth of patterns that a stochastic model can generate under the same set of parameters. When models of pattern formation become more and more detailed, they hit more and more of the same challenges faced by the underlying biological data that they represent.

Whether studying zebrafish patterns or another biological system, the first step to relating modeling frameworks at large scale is quantitatively describing self-organization. Methods for quantifying *in vivo* and *in silico* patterns include pair-correlation functions [26, 40, 95], order parameters [22, 25, 52], and pattern-simplicity scores [71, 72]. Recently tools from topological data analysis—particularly persistent homology—[17, 18, 19, 20, 29, 41, 74, 78] have also

emerged as a flexible means of quantifying complex systems and pattern formation [1, 94, 97]. Persistent homology has been applied to many biological systems, including flocking [7, 94], vascular networks [5, 76, 92], migrating cells [8, 11], intracellular dynamics [23], and fish skin [68].

Persistent homology characterizes connected components, holes, and higher-dimensional features in data across scales [78], and interpreting topological summaries can involve choices and hyper-parameters that are application specific. Recently McGuirl et al. [68] developed a pipeline for interpreting information from persistent homology as descriptions of spots and stripes in an agent-based model of zebrafish patterns [102]. The quantification results [68] involve several hyper-parameters, which are motivated by length scales in the model [102]. A related issue is that persistent homology is sensitive to outliers [4, 35], and this is challenging in biological settings, where noise is inherent. This raises questions about how McGuirl et al.'s approach can be adapted to quantify zebrafish patterns in noisier *in vivo* settings or alternative *in silico* settings. It is also unclear how sensitive the methods [68] are to the choice of hyper-parameters when interpreting topological summaries.

Motivated by the challenges associated with meaningfully relating different stochastic, microscopic models, here we apply methods from topological data analysis and computational geometry to quantify stripe patterns from two models of zebrafish at large scale. The two focal models of our case study consist of one off-lattice [102] and one on-lattice [80] individual-based model. These models [80, 102] are a useful place to center our work because they have largely the same cell interactions at their core from a biological perspective, but their computational implementation, cell density, and noise structure differ significantly. Drawing on these differences and similarities, we (1) interpret information from persistent homology [68] to estimate the number of stripes in patterns, compute stripe width, and identify stripe interruptions. We then (2) use clustering to group cells into pattern features, and (3) apply α -shapes, a tool from computational geometry [28, 30], to flexibly trace out stripe boundaries and estimate curviness. Each of these techniques relies on hyper-parameters, and we outline how to select their values in a way that ensures our comparisons are robust and fair.

Our work illustrates the challenges that arise when relating biologically similar—but mathematically and computationally different—stochastic models, and it shows how choices in the quantification process affect interpretations of results. This helps open the door to future large-scale studies of stochastic, spatial models, centered on questions including how sensitive model predictions are to implementation choices, and on how alternative mathematical perspectives can be combined to address biological questions.

2. Background and methods. We first provide a brief overview of zebrafish-pattern biology (subsection 2.1) and the two models [80, 102] that serve as the basis of our case study (subsection 2.2). Because our quantification pipeline relies on tools from topological data analysis and computational geometry, we then introduce persistent homology [68, 78] and α -shapes [28, 30] in subsection 2.3.

2.1. Biological background. Wild-type zebrafish are characterized by dark stripes and gold interstripes that form in their skin [55, 88, 99]. These patterns emerge from the interactions of tens of thousands of pigment cells arranged in three layers [51] (Figure 1(d)). There are three main types of cells involved in patterning: black melanophores (M), dense orange

(X^d) or loose yellow (X^ℓ) xanthophores, and dense silver (I^d) or loose blue (I^ℓ) iridophores [99]. The melanophores—present in the deepest layer—are restricted to dark stripes on the main fish body, whereas the other two cell types appear across the pattern in different colors and forms [37, 50, 51, 66, 70, 84]. Housed in the surface layer, X^d cells are tightly packed in interstripes, and X^ℓ cells (also called xanthoblasts) appear in a looser form at lower density in stripes [66]. Similarly, iridescent iridophores are present as I^d in interstripes and I^ℓ in stripes, sandwiched in the middle layer between melanophores and xanthophores in the skin [51, 91].

As wild-type zebrafish develop over a few months, all of these pigment cells interact through movement, division, death, and differentiation to form patterns. Cell-cell interactions may be mediated by diffusing signals [86], cellular extensions [32, 48, 53], or other mechanisms. In all cases, cells occupy a messy, biological environment, and there is inherent noise in their behavior. New pigment cells can appear through division of existing cells or by differentiation of precursors [12, 27, 47, 66, 70, 91]. Some cell behaviors are relatively well understood, and in other cases the interactions underlying cell dynamics are unknown [65, 104]. Iridophores, in particular, are a place in zebrafish-pattern biology where the field has been actively evolving [47, 65, 86, 87, 90, 91]. Up until roughly 2014, the focus was on melanophores and xanthophores, and iridophores were not heavily considered (e.g., [64, 84, 93, 108]). This changed with new observations [91] suggesting iridophores may swap their shape and color in specific ways (e.g., a loose blue iridophore transitioning to a silver iridophore). More recently empirical understanding of iridophores has continued to grow and shift, with Gur et al. [47] suggesting in late 2020 that iridophores differentiate in dense or loose form from precursors, rather than changing their shape once established.

Because our work applies persistent homology to understand cell organization, the length scales involved in zebrafish patterns are particularly important in this study. Roughly, the distance between nearest-neighboring cells tends to be around 50 micrometers (μm) [65, 85, 93], and, at the pattern scale, stripes and interstripes in adult fish are about 500 μm wide [34, 48, 102]; see Figure 4(a). As we discuss in subsection 2.2, the simulated patterns that we quantify correspond to pre-adult fish that are 12.63 millimeters (mm) long in “standard length” (SL), which refers to the distance from fish snout to where the body and tailfin meet (Figure 1(a)) [83]. Because zebrafish grow at different rates, it is customary to describe age in terms of standardized standard length (SSL, a measurement of SL associated with a reference zebrafish) or developmental stage, rather than time in days post fertilization [83]. When they reach 12.63 mm SSL, zebrafish are in the juvenile developmental stage; at this point, patterns generally include three light interstripes and two dark stripes. Measuring from the front edge of the bottom fin in Figure 1(a), this corresponds to a fish body that is about 2.3 mm high [83].

2.2. Focal mathematical models. Our case study centers on two closely related models that are stochastic and biologically detailed: the off-lattice model [102] developed by Volkening and Sandstede in 2018, and the on-lattice model [80] built by Owen, Kelsh, and Yates in 2020. For the remainder of this paper, we refer to the former as the ABM [102] and the latter as the CA [80]; see Figure 3. In terms of the wild-type cell interactions that they describe and predict, the models [80, 102] are largely the same from a biological perspective. They differ primarily in choices of computational implementation, which, in turn, affect how stochasticity shows up in

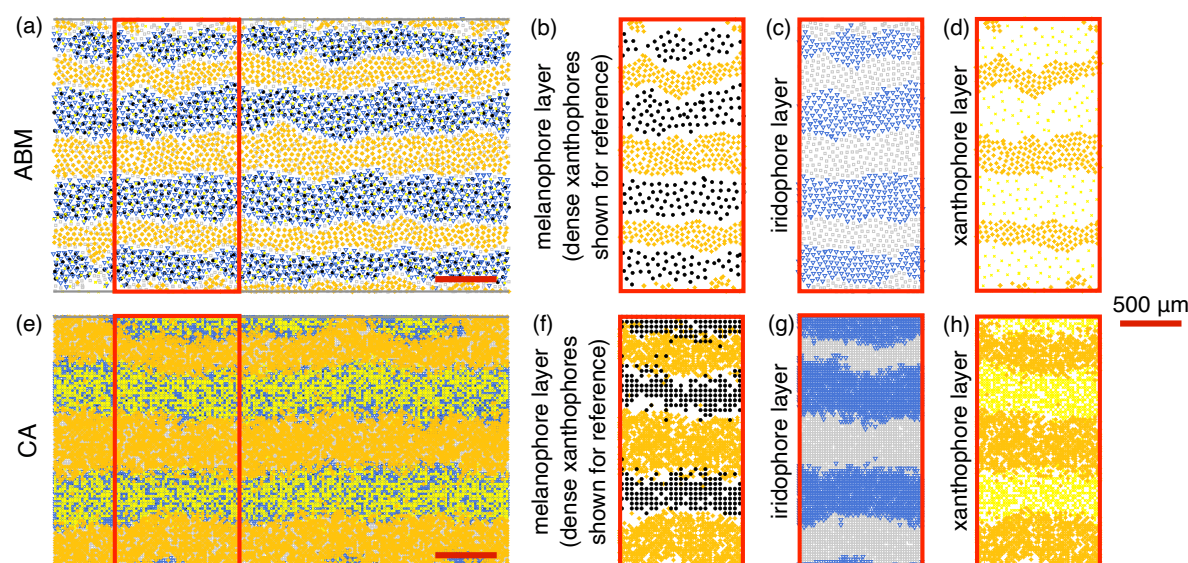


Figure 3. Patterns for our focal models [80, 102]. (a) The ABM [102] produces patterns with interstripe and stripe cells in distinct regions. (b) Cells occupy three layers in zebrafish skin (Figure 1(d)) [51], and we show these layers separately for easier viewing. (We include X^d cells with M cells for reference.) In the ABM [102], the presence of three layers of cells is taken into account indirectly by specifying volume-exclusion forces only between cells that occupy the same layer of skin. Orange X^d cells are separate from M cells. (c) Blue I^l and silver I^d cells occupy different regions in the ABM, as do (d) yellow X^l and orange X^d cells. (e) CA [80] patterns feature xanthophores and iridophores at higher density than ABM patterns, and (f) there is some overlap of M and X^d regions, (g) I^l and I^d regions, and (h) X^l and X^d regions. For comparison, we use the same symbols to plot cells in the ABM and CA patterns in (a)–(h), highlighting differences in cell density.

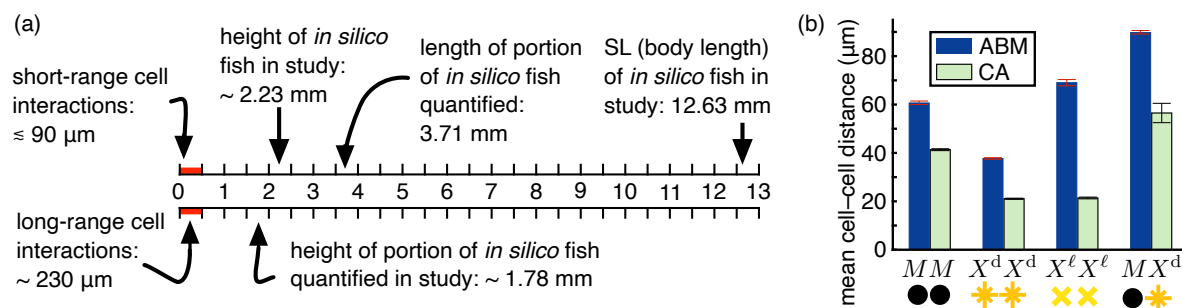


Figure 4. Biological length scales relevant to our study [80, 102]. (a) Length scales in our study range from less than 90 μm to over 10 mm [65, 80, 83, 85, 93, 102]. (b) We show the mean distance between nearest-neighboring cells, averaging across 1000 simulations [80, 102]. (The distance between M and X^d cells is for cells at stripe–interstripe boundaries: to approximate this, we take the mean distance between nearest M and X^d cells with distances greater than or equal to 110 μm or less than 0 μm excluded. This choice may overestimate the distance between M and X^d cells for the CA [80] slightly, since these cells can occupy the same position in different grids: if we consider distances of 0 μm , the mean distance between M and X^d cells decreases by about 2 μm .) Bars indicate the standard deviation in the mean distances.

cell behavior and pattern features. (We focus on the models [80, 102] because of their biological similarities and mathematical differences; in the future, it would be interesting to consider other models that differ in the types of cells that they consider, such as [13, 16, 59, 101].) Both models [80, 102] were built pre-publication of [47], in a biological environment in which the prevailing view was that iridophores may undergo shape and color changes. Because our focus is on developing methods for comparing similar models implemented in different ways, the new findings [47] do not affect our study. We overview the ABM and CA here; see Appendix A for a deeper dive into a few example model rules, and [80, 102] for full details.

The ABM [102] and CA [80] both include five cell types: black M , blue I^ℓ , and yellow X^ℓ cells in stripes, and silver I^d and orange X^d cells in interstripes. They also account for five dynamics: fish growth, along with cell movement, division or differentiation, death, and shape changes. The ABM [102] treats cells as particles interacting in continuous space, and tracks the positions (i.e., (x, y) -coordinates) of cells (point masses) interacting in growing domains. Volkening et al. implement cell movement through coupled differential equations: each pigment cell has an ODE describing the forces that it feels from other cells. Cell differentiation, division, death, and changes in shape—the dynamics that influence cell number—all take the form of stochastic, discrete-time rules. The ABM [102] accounts for three layers of cells in 2D domains indirectly by specifying volume-exclusion forces only between cells that occupy the same layer in the skin. On the other hand, the CA [80] is discrete in space, allowing cells to occupy positions in grids which grow in time through the addition of new grid squares. All cell behaviors—movement, differentiation, division, death, and shape changes—in Owen et al.'s model are based on stochastic, continuous-time rules, and they [80] include three separate 2D grids for melanophores, xanthophores, and iridophores.

Both approaches [80, 102] simulate pattern formation starting from the pelvic-bud developmental stage, when zebrafish are about 7.6 mm long in standard length and roughly 1.0 mm high. Volkening et al. [102] and Owen et al. [80] start with a rectangular domain that represents the full fish height (1 mm) and a portion of the fish body length. Because the models account for the remainder of the fish body differently, their domains grow at different rates. As we discuss in subsection 3.1, we focus on quantifying patterns that correspond to juvenile zebrafish at 12.63 mm in standard length, and this milestone is associated with ABM and CA patterns on domains of different sizes. Growth affects cell positions in both models: in the ABM, cell positions are multiplied by a scaling factor to account for uniform growth in a spatially continuous manner. In the CA, grid squares are inserted at randomly selected locations, modeling uniform growth in a spatially discrete way.

In addition to treating space as continuous or discrete, the models [80, 102] differ in how they treat time. The ABM has a time step of one day, and three main dynamics occur on each simulated day of fish development. First, the domain size is increased and cell positions are stretched to account for fish growth in a deterministic manner. Second, all of the cells undergo deterministic migration simultaneously. Third, the numbers and types of cells present are simultaneously updated to account for cell death, differentiation, division, and shape changes. For example, this means that every existing M cell is evaluated simultaneously for possible death (according to stochastic rules) each day. In comparison, the CA functions in continuous time, with an exponential clock that goes off when an “event” should occur. This event could be vertical domain growth, horizontal domain growth, evaluating a randomly selected M cell

for potential death, etc. Owen et al. [80] selected most rates for each of these events so that all of their cells are evaluated for that event once per day. This suggests that both models may be capturing the same number of behaviors per day on average. However, the ABM implements cell interactions simultaneously and the CA updates cell populations iteratively.

2.3. Geometric and topological data analysis. The microscopic models [80, 102] that serve as the basis of our case study produce point-cloud data in the form of pigment-cell locations. As we show in Figure 3, stripes are visible in patterns generated by these models [80, 102], but the discrete nature of the data makes them challenging to quantitatively describe. To help address this, our methodology for quantifying stripe patterns in section 3 draws on techniques from topological data analysis and computational geometry. Specifically, we utilize persistent homology and α -shapes, which naturally lend themselves to point-cloud data. We provide an informal introduction to persistent homology and α -shapes here; see [17, 20, 29, 41, 78, 94] and [28, 30], respectively, for more technical details.

2.3.1. Persistent homology. Persistent homology provides information about the number of connected components, holes, trapped volumes, and higher-dimensional features in data across scales [78]. Topaz, Ziegelmeier, and collaborators [7, 94, 97] have used persistent homology to understand the shape of complex-systems data, and topological techniques are being increasingly applied to *in vivo* and *in silico* biological systems [1]; examples include [7, 11, 23, 68, 76]. Simplicial complexes are at the core of persistent homology, and we focus on one type of simplicial complex, namely the Vietoris–Rips complex. To build a Vietoris–Rips complex, we start with point-cloud data, such as the positions of N cells in a domain. We place a ball with diameter ε around each point and connect two points with an edge whenever their respective $\varepsilon/2$ -balls intersect, indicating that the two points are at most a distance ε apart [7, 94]. If three points are each pairwise at most ε apart, we note this by filling in the triangle bound by the edges between pairs of points. Data points are 0-simplices, edges connecting two points are 1-simplices, and filled-in triangles are 2-simplices [7, 94]. We can continue this process to generate k -simplices for $k \geq 0$, but we focus on $k = 0$ and $k = 1$ here.

By varying ε , we can study how the shape of our point-cloud data evolves across scales. Specifically, we are interested in the presence of k -dimensional holes, meaning empty regions bound by k -simplices [7, 94]. Connected components are 0-dimensional holes, and 2D trapped areas are 1-dimensional holes (i.e., loops) bound by 1-simplices. As we allow ε to grow, the shape of our data in dimension 0 evolves from N isolated points when $\varepsilon = 0$ to one large connected component once ε is large enough. McGuirl et al. [68] interpreted dimension-1 topological features, in particular, as holding information about stripe patterns generated by the ABM [102]. Because the ABM [102] has periodic boundary conditions in x , pattern formation is occurring on a cylinder, as we show in Figure 5(b). If we apply persistent homology to the positions of X^d cells on this cylindrical domain, interstripes are visible as loops in our data for a range of ε values. On the other hand, applying persistent homology to the positions of M or X^ℓ cells translates into information about stripes [68].

To make this relationship between (inter)stripes and dimension-1 topological features more precise [68], it is useful to discuss birth and death times [94]. The “birth time” of a topological feature is the value $\varepsilon = \varepsilon_{\text{birth}}$ at which the feature first appears, and the “death time” is the value $\varepsilon = \varepsilon_{\text{death}}$ at which the feature disappears [68]. A feature’s “persistence” is $\varepsilon_{\text{death}} - \varepsilon_{\text{birth}}$,

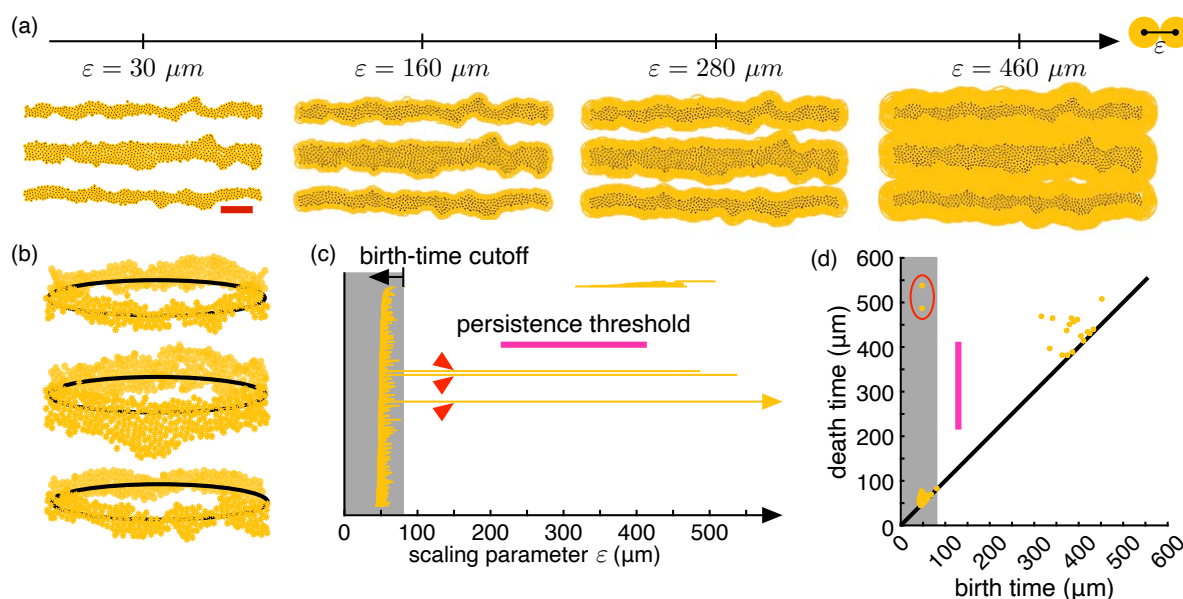


Figure 5. Persistent homology of zebrafish patterns [68]. (a) As an example, we show gold balls of varying diameter ε centered at the positions of X^d cells (black points here) from the ABM [102]. (b) Boundary conditions are periodic in x for the models [80, 102], so unbroken stripes and interstripes form loops. (c) We show the barcode describing the dimension-1 topological features associated with the pattern in (a). McGuirl et al. [68] interpreted bars with birth times less than a birth-time cutoff (80 μm) and persistence greater than a threshold (200 μm) as corresponding to interstripes. There are three long bars (marked by red triangles) that meet these conditions, corresponding to three clear interstripes in (a). (d) Persistence diagrams are an alternative means of summarizing information from persistent homology; here each point represents the birth and death times of a loop. Because the final loop in (a)—a function of our periodic domain—persists longer than the maximum ε value that we considered, only two persistent loops (circled in red) are visible.

and this information is often visualized using barcodes and persistence diagrams. We show the barcode and persistence diagram associated with dimension-1 topological features for an example of three interstripes in Figure 5. In a barcode, we draw a horizontal bar starting at $\varepsilon_{\text{birth}}$ and ending at $\varepsilon_{\text{death}}$ for each topological feature [94]; these bars are stacked vertically. A persistence diagram is a scatter plot with birth time as the x -axis and death time as the y -axis; each point $(\varepsilon_{\text{birth}}, \varepsilon_{\text{death}})$ refers to the birth and death times of a feature. As we show in Figure 5(c), there are three long bars, corresponding to three highly persistent loops. Specifically, McGuirl et al. [68] defined the number of interstripes in ABM [102] patterns by counting the number of X^d loops with persistence greater than a persistence threshold, and birth time less than a birth-time cutoff. The persistence threshold and birth-time cutoff are hyper-parameters. In subsection 3.3, we build on the work [68] to make this approach more flexible for on-lattice stripe patterns and reliably choose hyper-parameter values.

2.3.2. Alpha-shapes. Complementing information about the number of connected components and holes in data that persistent homology can provide, α -shapes, a concept from computational geometry, are a flexible means of characterizing the overall shape of data [28]. A generalization of the convex hull rigorously defined by Edelsbrunner et al. [30], α -shapes

have been applied to reconstruct surfaces [46, 107], as well as estimate perimeter [2], surface curvature [61], shape complexity [39], and volume [62]. For example, α -shapes have been used in structural biology [62] to better understand the shape of protein structures [109]. Like persistent homology, α -shapes lend themselves to point-cloud data (i.e., cell coordinates in our case study) and provide information about shape. They also naturally involve the choice of a hyper-parameter α , and this inherent flexibility makes them suitable for our study.

Alpha-shapes are often described intuitively using circular erasers or spoons with radius α [28, 30]. In this analogy [28, 30], constructing an α -shape is thought of as carving away from the convex hull of a point cloud whenever it is possible to do so without removing any data point. At the final stage, all of the eraser marks with curvature $1/\alpha$ are approximated as straight edges. In particular, given some set of points, an associated α -shape is made up of points and edges representing the boundary of the point cloud [30]. Depending on the value of the hyper-parameter α , we can select how tightly the α -shape fits our data. For finite $\alpha > 0$, the associated α -shape is built by drawing an edge between two data points whenever we can place a ball of radius α in such a way that (1) the open ball contains no points in our data set, and (2) the two data points in question lie on the boundary of the closed ball [28].

In Figure 6(a), we show an example point cloud, consisting of discrete points in the shape of an overturned dumbbell. When α is large (e.g., $\alpha \approx 50$ in Figure 6), the boundary of the α -shape is nearly the convex hull of our data, and we miss meaningful structure in the dumbbell shape. Indeed, when $\alpha = \infty$, the α -shape is simply the convex hull of the point cloud. As we decrease α in Figure 6(b), our α -shapes look more like hourglasses and eventually pick up the shape of the dumbbell between $\alpha = 10$ and $\alpha = 5$. When $\alpha = 0$, the α -shape is just the data itself [31]. (Notably, α -shapes are also defined for $\alpha < 0$; see [30, 31] for more information.) While we do not discuss it further here, there is a connection between α -shapes, Voronoi diagrams, and Delaunay triangulations [28, 30, 62]. Additionally, α -shapes are related to α -complexes, which—like the Vietoris–Rips complex that we introduced in subsection 2.3.1—can serve as the basis for computing persistent homology [28, 31, 78]. In subsection 3.5, we apply α -shapes to extract stripe–interstripe boundaries and quantify pattern curviness.

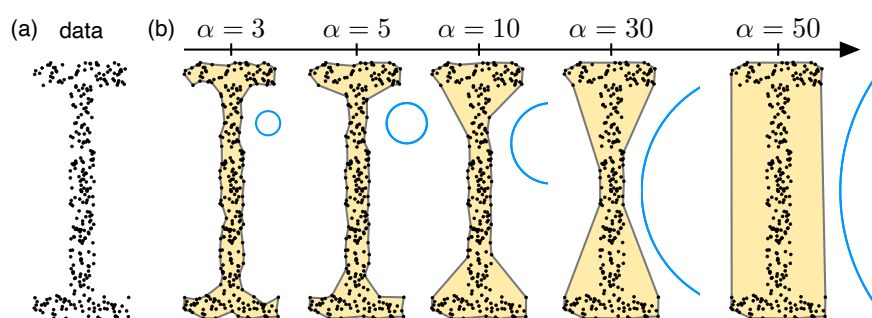


Figure 6. Introduction to α -shapes [28, 30]. (a) As an example, we consider point-cloud data in the shape of a dumbbell turned on its side. (b) Alpha-shapes [28, 30], a generalization of the convex hull, are a flexible means of characterizing the boundary of point-cloud data. As $\alpha \geq 0$ gets larger, we estimate the boundary of data less and less tightly, approaching the convex hull as $\alpha \rightarrow \infty$. We show the circle (or portion of the circle) with radius α used to construct each α -shape in blue.

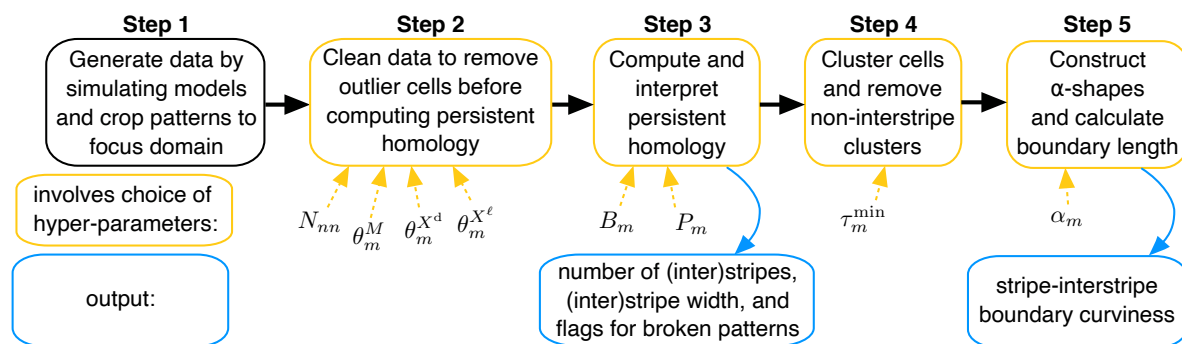


Figure 7. Summary of our quantification process and hyper-parameters. To generate our data, we simulate both models [80, 102] and crop the final patterns to a focus domain length. Because persistent homology can be sensitive to outliers [4, 35] and CA patterns [80] often feature stray cells, we perform an initial cleaning step before computing persistent homology using the distances between cells on a domain that is periodic in x (see Figure 5(c)). We use the birth times and persistence of dimension-1 topological features—corresponding to loops—to identify the number of stripes and interstripes in each pattern, flag broken patterns, and estimate the maximum width of (inter)stripes [68]. Before constructing α -shapes [28, 30] and extracting their boundaries to quantify interstripe curviness, we cluster X^d cells into the number of interstripes that we identified in Step 3. If a cluster has fewer than τ_m^{\min} cells, it is likely a stray cell or spot, and we remove it and re-cluster, further cleaning the data. Importantly, hyper-parameters enter our methods in Step 2 (e.g., defining “outlier”), Step 3 (i.e., our birth-time cutoff multiplier and persistence threshold), Step 4 (i.e., τ_m^{\min}), and Step 5 (i.e., our alpha radius). As we discuss in subsection 3.2, we use $N_{nn} = 10$ cells, $\theta_{ABM}^M = \theta_{CA}^M = 220 \mu\text{m}$, $\theta_{ABM}^{X^d} = \theta_{CA}^{X^d} = 130 \mu\text{m}$, $\theta_{ABM}^{X^l} = 215 \mu\text{m}$, and $\theta_{CA}^{X^l} = 170 \mu\text{m}$ for all of our results. Based on our robustness study in subsection 4.1, we set $B_{ABM} = B_{CA} = 2.5$ mean cell-cell distances as our baseline birth-time cutoff; $P_{ABM} = P_{CA} = 200 \mu\text{m}$ as our persistence threshold; $\tau_{ABM}^{\min} = \tau_{CA}^{\min} = 60$ cells as our minimum cluster size; and $\alpha_{ABM} = \alpha_{CA} = 100 \mu\text{m}$ as our alpha radius. We use these baseline hyper-parameter values for our results in subsection 4.2.

3. Results: Our methodology for quantifying stripe patterns. Here we develop a pipeline for quantifying stripe patterns from the off-lattice model [102] and the on-lattice model [80] introduced in subsection 2.2. Our methodology involves cleaning the pattern data to remove outlier cells, computing persistent homology using cell coordinates, building on the methods [68] to interpret persistent homology more flexibly, and constructing α -shapes to characterize curviness. Our five-step process in Figure 7 allows us to comprehensively quantify stripe patterns emerging from stochastic on- and off-lattice models in a robust, comparable way. As we summarize in Figure 8, we show how to count the number of stripes and interstripes, detect the presence of pattern breaks, quantify (inter)stripe width, and characterize interstripe curviness. Throughout subsections 3.1–3.5, we pay particular attention to pointing out hyper-parameters and choices in our analysis pipeline.

In addition to providing a broader perspective on wild-type zebrafish stripes *in silico*, the summary statistics that our methods produce and the flexibility of our approach open the door to future studies of several mutants. For example, compared to wild-type zebrafish, *jaguar/obelix* [56, 64] mutants feature widened stripes or interstripes. The stripes in some *leopard* and *luchs* phenotypes [54, 64, 103] are undulating or curvy, and *idfix* [36] zebrafish typically have narrower stripes with breaks or interruptions. As more *in vivo* and *in silico* data on these mutants become available, we expect that our methods could be used to detect and quantitatively characterize the detailed differences between these striped phenotypes at

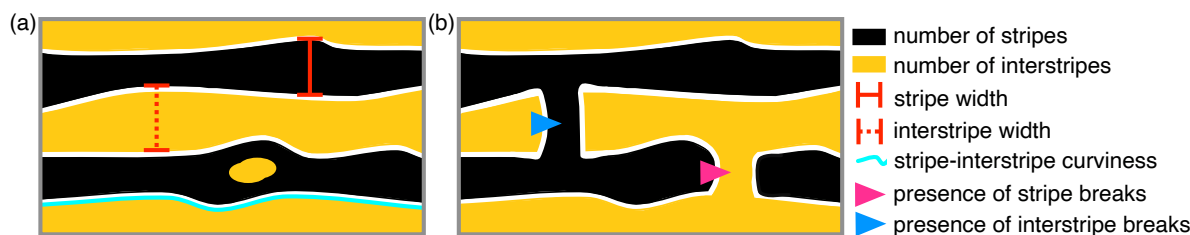


Figure 8. Summary of the pattern features that our methods characterize: number of (inter)stripes, (inter)stripe width, stripe–interstripe boundary curviness, and interruptions in (inter)stripes. See section 3 and Figure 7. (a) In order to compute (inter)stripe width and stripe–interstripe boundary curviness, we first need to identify the number of stripes and interstripes in each pattern. (b) Some patterns feature breaks or interruptions in (inter)stripes, and our methods detect when breaks are present. While our focus is on wild-type simulations from the off-lattice model [102] and the on-lattice model [80], being able to quantify these features of stripe patterns has broader biological value. In particular, several zebrafish mutants are characterized by stripe patterns that are wider, curvier, or more frequently broken than wild-type patterns [36, 54, 64, 103].

large scale. To support future research directions like these, the code that we developed to quantify patterns, as well as sample data, is publicly available on GitLab [24].

3.1. Step 1: Generate and format data. To generate the data for our case study on zebrafish patterns, we simulate the ABM [102] and CA [80] under wild-type conditions. This step involves mild post-processing to frame the output of both models in the same way. As we discussed in subsection 2.1, zebrafish development is generally measured in fish length, rather than in time, so we simulate these models until a standard length of 12.63 mm. Because the ABM [102] includes deterministic domain growth, this corresponds to simulating the model for 45 days, until the fish reaches an age of 66 days post fertilization. In contrast, the CA [80] features stochastic domain growth, so there is not a direct correspondence between simulation time, domain size, and fish standard length. Instead, we simulate the CA [80] until the domain reaches a size associated with a standard length of 12.63 mm.² We simulate both models 1000 times and save the results at the final time.

The positions of cells in the ABM [102] are in the form of (x, y) -coordinates in μm , while the positions of cells in the CA [80] are described using lattice indices. We frame the output of both models as coordinates by using that Owen et al. [80] assume their xanthophore and iridophore grid squares are $20 \mu\text{m}$ wide, and their M grid squares are $40 \mu\text{m}$ wide (e.g., we map the indices (i, j) of a melanophore to (x, y) -coordinates $(40i, 40j)$). At this stage, all of the ABM [102] patterns are on domains of size 3.71 mm long and 2.215 mm high. The domains associated with CA [80] patterns are all within one grid step of 6.93 mm in length, and they vary from 1.56 to 3.08 mm in height (in our 1000 simulations), since growth is uncoupled in the horizontal and vertical directions. To account for the difference in domain length between the models, we crop CA patterns, only considering the cells in a region of length 3.71 mm in the center of the domain. Following the approach in [68], we remove the cells that fall within the top 10% or bottom 10% of the domain height for both the ABM and CA data, since stripes are often messy and not fully formed in those regions.

²Because the CA model [80] works on a discrete domain, the final domain size is within one grid square (maximally $40 \mu\text{m}$) of the target length.

We crop patterns vertically using a percentage—rather than a strict number—based on our observations that both models have been developed to produce the same number of stripes and interstripes at a standard length of 12.63 mm. In the ABM [102], the result is patterns on domains of cropped height 1.772 mm. CA [80] cells fall within domains of height between 1.248 and 2.464 mm, depending on the simulation, after our cropping step. (When CA domains are taller, stripes and interstripes are also wider.) We show example patterns highlighting the cells that have been cropped out from the top and bottom of the domains in Figure 9. At the end of Step 1, we have formatted the output of both models as coordinates of cell positions in micrometers on domains of length 3.71 mm and variable height.

3.2. Step 2: Clean data to remove outlier cells. Our observations suggest that stripe and interstripe cells occupy distinct regions in ABM [102] patterns. In comparison, the CA [80] frequently produces outlier cells, such as an isolated black cell in the center of a gold

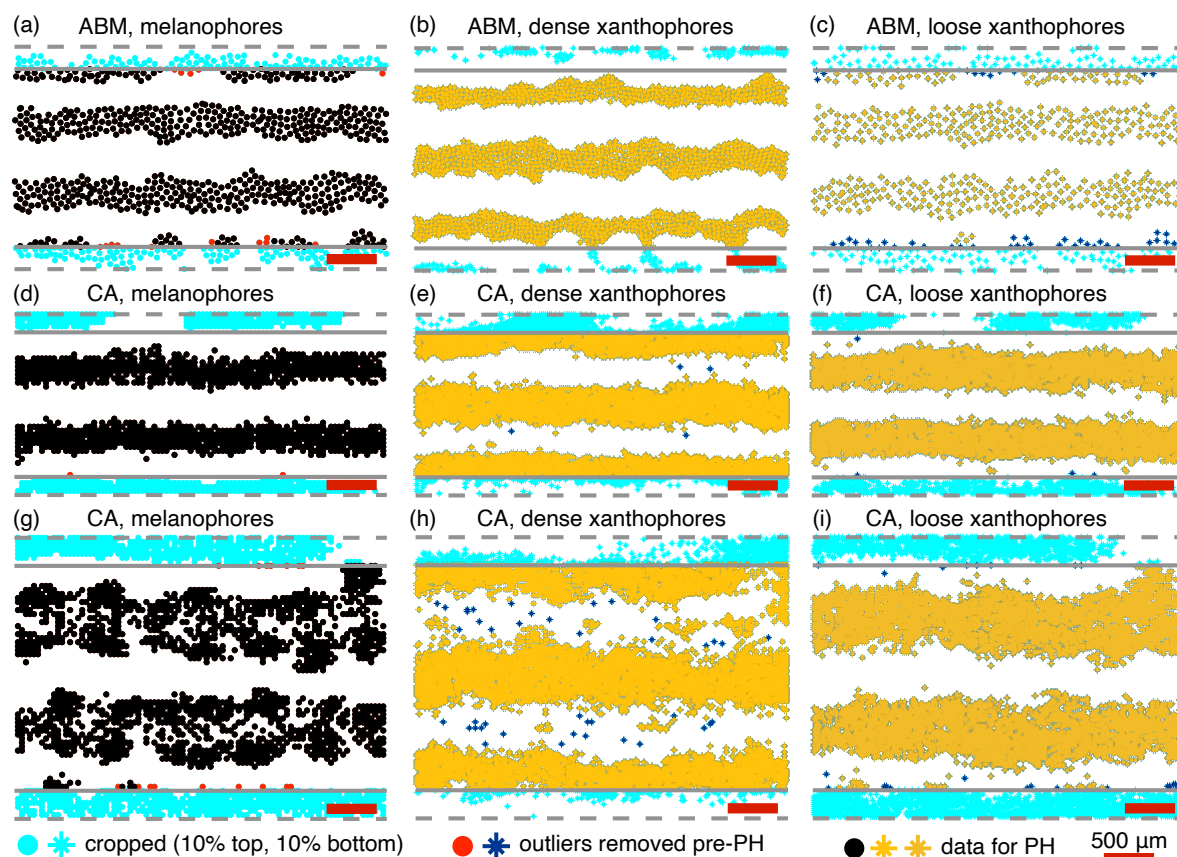


Figure 9. Steps 1 and 2: Cropping and cleaning to remove outliers. For an example ABM [102] pattern in (a) and two example CA [80] patterns in (d), (g), we indicate the M cells that we crop out in Step 1 (cyan) and remove as outliers in Step 2 (red). For the same simulations, the (b), (e), (h) X^d and (c), (f), (i) X^t cells that we crop out are cyan, and those we remove as outliers are dark blue. The gray dashed bars are the domain boundaries, and the solid gray bars mark the cropped boundaries. See Figure 10 for corresponding summary statistics.

interstripe; see Figure 3. Because the way that we compute persistent homology in Step 3 is sensitive to outliers like these [4, 35], we perform an initial cleaning step to remove stray cells. Focusing on the three cell types— M , X^d , and X^ℓ —for which we compute persistent homology, we identify outliers by first finding the $N_{nn} = 10$ nearest neighbors of each cell under the Euclidean norm on a domain that is periodic in x . We only consider neighbors of the same cell type, and we exclude each cell from counting itself. Using the distances $\{d_i\}_{i=1,\dots,N_{nn}}$ from the j th cell of type $c \in \{M, X^d, X^\ell\}$ to its nearest neighbors of type c , we compute $D_j^c = \sqrt{\sum_{i=1}^{N_{nn}} d_i^2 / N_{nn}}$. If $D_j^c \geq \theta_m^c$, where $m \in \{\text{ABM}, \text{CA}\}$, then we classify the j th cell of type c as an outlier and remove it. This captures that outliers are in low-density areas, so their distance from their ten nearest neighbors is comparatively large.

We use the thresholds $\theta_{\text{ABM}}^M = \theta_{\text{CA}}^M = 220 \mu\text{m}$ to identify melanophore outliers; $\theta_{\text{ABM}}^{X^d} = \theta_{\text{CA}}^{X^d} = 130 \mu\text{m}$ to highlight dense xanthophore outliers; and $\theta_{\text{ABM}}^{X^\ell} = 215 \mu\text{m}$ and $\theta_{\text{CA}}^{X^\ell} = 170 \mu\text{m}$ to indicate loose xanthophore outliers. These initial cleaning thresholds, as well as our choice to take into account a cell's $N_{nn} = 10$ nearest neighbors, are hyper-parameters that we do not investigate further. We chose them by observation, selecting values that identified outliers and fell in a range insensitive to changes. The larger distance thresholds for melanophores reflect that the distance between these cells is larger; see Figure 4(b). We show the distributions for the fraction of cells removed by type and model in Figures 10. The number of M cells cleaned is comparable: a mean of 13.6 cells with a standard deviation of 6.9 cells for the ABM, and a mean of 14.8 cells with a standard deviation of 8.3 for the CA, across our 1000 simulations. In comparison, the numbers of X^d and X^ℓ cells removed differ between the models. On average, our methods identify zero X^d outliers in the ABM [102], and 22.6 X^d outliers with a standard deviation of 9.2 in the CA [80]. For X^ℓ cells, we find a mean and standard deviation of 48.1 ± 9.4 outliers in the ABM and 15.2 ± 8.4 outliers in the CA.

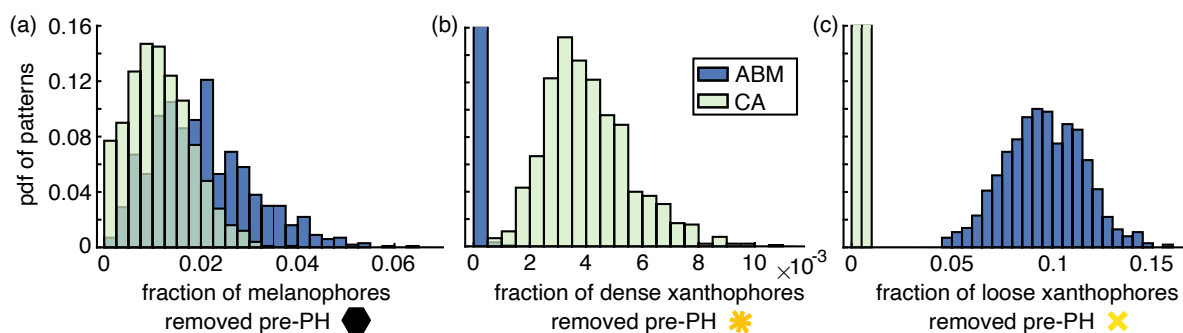


Figure 10. Step 2: Cleaning to remove outliers. Across 1000 ABM [102] and CA [80] simulations, we show distributions for the fraction of (a) M , (b) X^d , and (c) X^ℓ cells that we identify as outliers and remove in Step 2 before computing persistent homology (PH). On average, we remove about 1.16% and 2.02% of CA and ABM M cells, respectively; 0.41% and 0.00% of CA and ABM X^d cells, respectively; and 0.34% and 9.59% of CA and ABM X^ℓ cells, respectively. Importantly, the places in the domain where cells are being identified as outliers differ between CA and ABM patterns; see Figure 9 and subsection 3.2. Fractions are based on the cells present after cropping in Step 1. See Supplementary Material (M154308_01.pdf [local/web 1.04MB]) for corresponding distributions for the number of cells removed in Step 2; the number of cells removed due to cleaning ranges from roughly 0 to 80 cells.

The places where cells are being identified as outliers in the two models are also different; see Figure 9. As a result, cleaning the ABM data does not affect pattern quantification, and we only do this to apply the same set of steps to both models. We stress that McGuirl et al. [68] showed that persistent homology can be applied to quantify the ABM [102] without any cleaning. This is because the ABM does not produce interstripe cells in stripe regions and vice versa; instead, as we show in Figures 9(a)–(c), the “outliers” removed from ABM patterns tend to be cells at the upper and lower domain boundaries. In comparison, cleaning the CA data allows us to remove stray cells that occur between stripes and interstripes; see Figures 9(d)–(i). Without this cleaning, the presence of outlier cells in CA patterns makes interpreting topological summaries difficult in Step 3 (subsection 3.3). It is also worth noting that our initial cleaning does not account for the presence of X^d spots (see, for example, Figure 9(h)) in some CA stripes. These spots have high enough local density that they are not identified as outliers, and we return to handling these pattern features in Step 4 (subsection 3.4).

3.3. Step 3: Compute persistent homology to define stripe number and width. By building on the approach [68] (see subsection 2.3), we use dimension-1 topological features, which correspond to loops or holes in our data, to count stripes and interstripes, define stripe and interstripe width, and flag broken or incomplete stripes and interstripes. Using the cropped, cleaned data from Step 2, we calculate pairwise distances between M cells, between X^d cells, and between X^ℓ cells on a domain that is periodic in x . (Because we crop the CA patterns so that our CA and ABM domains have the same length in Step 1, we note that specifying boundary conditions that are periodic in x on these cropped domains is an approximation for CA patterns.) We compute persistent homology in dimension 0 and dimension 1 based on these pairwise distances using Ripser [3], enforcing a maximum distance of $650 \mu\text{m}$ in our filtration.

Specifically, we identify unbroken stripes and interstripes using the dimension-1 topological features (loops) for X^ℓ and X^d cells, respectively, as below:

$$(3.1) \quad \text{number of unbroken stripes} = \sum_{X^\ell \text{ loops } i} \mathbf{1}_{b_i < B_m \Delta_{X^\ell X^\ell}^m}(i) \cdot \mathbf{1}_{d_i - b_i > P_m}(i),$$

$$(3.2) \quad \text{number of unbroken interstripes} = \sum_{X^d \text{ loops } i} \mathbf{1}_{b_i < B_m \Delta_{X^d X^d}^m}(i) \cdot \mathbf{1}_{d_i - b_i > P_m}(i),$$

where (b_i, d_i) are the (birth, death)-coordinates of the i th loop; $d_i - b_i$ is the persistence of this feature; Δ_{cc}^m is the average distance between neighboring cells of type c for the model $m \in \{\text{ABM}, \text{CA}\}$ across all 1000 simulations (see Figure 4(b)); and the indicator function $\mathbf{1}_{\text{condition}}(i) = 1$ if the given condition is met for topological feature i , and 0 otherwise. The birth-time cutoff factor B_m and persistence threshold P_m with $m \in \{\text{ABM}, \text{CA}\}$ are our hyper-parameters for selecting the loops corresponding to stripes or interstripes in our data. In comparison, McGuirl et al. [68] used a strict birth-time threshold B in place of our more flexible cutoff $B_m \Delta_{cc}^m$ that depends on the mean distance between cells. We sweep across these hyper-parameters in subsection 4.1 to identify choices that are robust for our focal models.

Following the approach [68], we use M cells, in addition to X^d and X^ℓ cells, when flagging patterns with stripe or interstripe breaks. Because we expect three interstripes in simulated zebrafish at the stage that we consider, we count a pattern as having broken interstripes if its

number of interstripes in (3.2) is less than three. We classify a pattern as having broken stripes if two conditions are met: (1) its number of stripes in (3.1) is less than two (the expected number of stripes for our simulated fish) and (2) its number of unbroken stripes from the perspective of melanophores is also less than two, where we define

$$(3.3) \quad \text{number of stripes from } M \text{ perspective} = \sum_{M \text{ loops } i} \mathbf{1}_{b_i < B_m \Delta_{MM}^m(i)} \cdot \mathbf{1}_{d_i - b_i > P_m(i)}.$$

McGuirl et al. [68] used both M and X^ℓ cells to identify broken stripes in the ABM [102] because Volkening et al.'s model produces X^ℓ cells at low density; see Figure 3(d). This rationale applies in the reverse to the CA [80]: X^ℓ cells appear at high density in interstripes, whereas M cells are at comparatively low density, particularly in wide stripes; see Figures 3(h) and 9(g). By flagging patterns as broken stripes only when the number of persistent loops is less than our expected number of stripes for both M and X^ℓ cells, we allow for local dips in density for some stripe cells, as long as another type of stripe cell remains at consistently high enough density. We choose to focus on xanthophores and melanophores throughout our stripe-quantification pipeline; in the future, it would also be interesting to include iridophores.

In addition to counting (inter)stripes, we use persistent homology to compute stripe and interstripe width. Broadly, the death times of topological loops associated with interstripe cells provide information on stripe width, and vice versa, as below:

$$(3.4) \quad \text{stripe width} = \underbrace{\text{med}_{X^d \text{ interstripe loops } i} d_i}_{\text{median death time for interstripe loops}} - \underbrace{\Delta_{MX^d}}_{\text{boundary separation}},$$

$$(3.5) \quad \text{interstripe width} = \underbrace{\text{med}_{X^\ell \text{ stripe loops } i} d_i}_{\text{median death time for stripe loops}} - \underbrace{\Delta_{MX^d}}_{\text{boundary separation}},$$

where we define X^d and X^ℓ dimension-1 topological features as “interstripe loops” or “stripe loops” using (3.2) and (3.1), respectively, and exclude features with infinite death times d_i . Here Δ_{MX^d} is the mean distance between nearest melanophores and dense xanthophores at the stripe–interstripe interface for the specific simulation in question; see Figure 4(b). Subtracting Δ_{MX^d} prevents us from counting the interface-boundary distance twice. Our measurements of the width of any given (inter)stripe should be interpreted as maximum width.³ As an illustrative example of this, in Figure 5(a), the width of the dark stripe between the bottom two interstripes is slightly larger than $460 \mu\text{m} - \Delta_{MX^d}$, since a small amount of white space (a hole) is still visible in between the interstripes at $\varepsilon = 460 \mu\text{m}$.

3.4. Step 4: Cluster cells into interstripes and further clean data. Broadly, we quantify curviness by clustering X^d cells into interstripes in Step 4 and then tracing out the upper and lower boundaries of each interstripe in Step 5. We use the method introduced in [68] to specify the number of interstripe features (i.e., the target number of clusters N_{cl}) and apply single-linkage hierarchical clustering to the positions of X^d cells to produce N_{cl} clusters. In the

³In comparison, McGuirl et al. [68] defined stripe width using the persistence of X^d loops. This approach [68] is similar to ours but does not account for the fact that the distance between M and X^d cells is larger than the distance between like cells. Additionally, due to an error in ball radius versus diameter, the summary statistics in [68] report double (inter)stripe width.

ABM [102], these clusters generally correspond directly to interstripes. Because of differences in the noise and messiness present in ABM [102] and CA [80] patterns, however, we find that the same process does not transfer to the on-lattice model [80]. Instead, as in the CA pattern in Figure 9(h), gold spots or stray X^d cells that remain after our initial cleaning in Step 2 can disrupt clustering. To account for these challenges, we introduce an iterative process to detect and remove clusters of X^d cells that do not correspond to interstripes. This is a second round of data cleaning, removing more cells beyond those that we already removed in Step 2.

Before identifying clusters corresponding to interstripes, the first step is determining the number of interstripe features. To account for gold bridges connecting broken stripes, McGuirl et al.'s method [68] clusters X^d cells into $N_{cl} = 3$ groups when a pattern has no breaks in black stripes; $N_{cl} = 2$ groups when the pattern has one broken stripe; and $N_{cl} = 1$ group otherwise. Our observations suggest that the models [80, 102] are meant to produce patterns with two stripes and three interstripes at the stage that we quantify. Similar to [68], we thus consider a pattern to contain one broken stripe when the number of stripes (from the perspective of X^ℓ or M cells—or from the perspective of both) is one; see (3.1) and (3.3). As a book-keeping step, we also sort our matrix of cell coordinates by their x -coordinates before performing any clustering. Having cell coordinates in order based on their position from the left to right boundary of the domain is convenient later in Step 5 (subsection 3.5).

In an iterative process, we cluster our sorted X^d cells into N_{cl} groups, applying single-linkage hierarchical clustering to cell coordinates. (We use the Euclidean distance on a domain that is periodic in x .) We next calculate the number of cells in each cluster, and, if the number of cells is strictly less than our minimum cluster size τ_m^{\min} with $m \in \{\text{ABM}, \text{CA}\}$, we remove all of the cells in that cluster from the domain. We then re-cluster the remaining X^d cells into N_{cl} groups, count the number of cells in our clusters, remove small clusters, and loop through again as needed. At the end of this process, the result is N_{cl} clusters of X^d cells that we interpret as corresponding to interstripes. We investigate the role of the hyper-parameter τ_m^{\min} in subsection 4.1.

3.5. Step 5: Construct α -shapes to quantify interstripe curviness. To compute the curviness of stripe–interstripe interfaces, the first step is extracting the boundaries of our clusters of X^d cells from Step 4. There are many ways to do this: for example, focusing on one interstripe at a time, McGuirl et al. [68] discretized space and then found the cells with the maximum and minimum y -coordinates in each grid step. Taken across all the grid positions, these highest and lowest cells trace out the upper and lower boundaries of an interstripe [68]. Owen et al. [80] developed an algorithm for extracting the upper and lower boundaries of the center interstripe in the domain. This approach [80] involves expressing the lattice of X^d cells as a binary image and applying the built-in MATLAB function *bwmorph* to clean the data and bridge small gaps in interstripes.⁴ Owen et al. then identify the height of the pixels associated

⁴The function *bwmorph* is a flexible means of adjusting binary images (matrices of zeros and ones) in MATLAB. Owen et al. [80] applied *bwmorph* with the built-in “clean” and “bridge” options. The “clean” option removes isolated one-valued pixels (i.e., sets them to zero-valued in the binary image); effectively, this means removing X^d cells without other X^d cells next to them in the lattice [80]. The “bridge” option fills in gaps by changing zero-valued pixels to one-valued when they appear next to two disconnected one-valued pixels; effectively, this means adding X^d cells to the lattice to fill in holes and smooth out gaps in the pattern [80].

with the upper and lower boundaries of the center interstripe. Their methods [80] focus on interstripe curviness as a whole—rather than stripe–interstripe boundary curviness—so they later take the mean of these two curves and apply a smoothing algorithm. Importantly, both approaches [68, 80] rely on discretizing space: the lattice structure of the CA [80] lends itself to image-processing techniques for binary images, and McGuirl et al. [68] introduce a spatial discretization to extract interstripe boundaries in the ABM.

We adapt α -shapes [28, 30] to develop a more flexible method for quantifying stripe–interstripe curviness in zebrafish patterns. As we discussed in subsection 2.3.2, α -shapes provide a means of describing the boundary of a point cloud and are a generalization of its convex hull; see Figure 6. Here our point cloud consists of the coordinates of X^d cells associated with each interstripe cluster from Step 4. One interstripe at a time, we construct an α -shape and extract the coordinates of the X^d cells lying on its boundary using the built-in MATLAB functions *alphaShape* and *boundaryFacets* (with holes suppressed). This leads to a sequence of coordinates $(\mathbf{X}_i^{d,\text{bnd}})_{i=1,\dots,N^{\text{bnd}}}$ that are the vertices of our α -shape, wrapping around the interstripe. We next find the first index k in this sequence with x -coordinates equal to the maximum x -coordinates in the sequence. Because we sort the list of X^d cells with respect to their x -coordinates in Step 4 before clustering into interstripes, this index splits the boundary X^d cells in a way that captures the top and bottom boundaries of the interstripe. Specifically, we consider the sequences $(\mathbf{X}_i^{d,\text{bnd}})_{i=1,\dots,k}$ and $(\mathbf{X}_i^{d,\text{bnd}})_{i=k,\dots,N^{\text{bnd}}}$. To complete the process, we remove the cells in both of these sequences that fall within $100\ \mu\text{m}$ (inclusive) of the right or left boundaries of the domain and renumber indices as needed.⁵ This last step ensures that we have excluded the right and left boundaries of our X^d point cloud. The result is two sequences of cell coordinates tracing out the top and bottom boundaries of the interstripe in question.

For each sequence of cell coordinates $(\mathbf{X}_i^{d,\text{bnd}})_{i=1,\dots,N}$ representing the boundary of an interstripe, we compute the length of the associated piecewise linear curve. We then use the length of a perfectly straight interstripe to compute curviness, similar to [68], as below:

$$(3.6) \quad \text{curviness of boundary} = 100 \times \left(\frac{\sum_{i=1}^{N-1} \|\mathbf{X}_i^{d,\text{bnd}} - \mathbf{X}_{i+1}^{d,\text{bnd}}\|}{\max_i(x_i^{d,\text{bnd}}) - \min_i(x_i^{d,\text{bnd}})} - 1 \right),$$

where $\|\mathbf{x}\| = \sqrt{x_1^2 + x_2^2}$ and $\mathbf{X}_i^{d,\text{bnd}} = (x_i^{d,\text{bnd}}, y_i^{d,\text{bnd}})$. Repeating this process for each of the N_{cl} interstripe clusters from Step 4, we generate $2 \times N_{\text{cl}}$ measurements of boundary curviness. Across all of these boundaries, we exclude the curves with the maximum and minimum y -coordinates, since we find that interstripes in some CA patterns (e.g., see Figure 9) extend beyond the limits of our focal domain. As a result, these uppermost and lowermost curves are often simply tracing the upper and lower boundaries of the rectangular domain. For the remaining $2N_{\text{cl}} - 2$ boundary curves, we take the mean of their curviness values from (3.6) and define this as our measurement of pattern curviness.

⁵Using $100\ \mu\text{m}$ here is a choice that we do not investigate further. Excluding this small region near the right and left boundaries of the pattern is fairer to both models. In particular, because we crop CA domains lengthwise in Step 1, they could potentially be at a disadvantage when computing curviness, since ABM domains have the benefit of unaltered periodic boundary conditions in x . In addition to helping us separate cell coordinates into upper and lower boundaries, cropping the region of the pattern for which we compute interstripe curviness by $200\ \mu\text{m}$ puts both models on equal footing.

Our method for going from α -shape vertices to two sequences of cells representing the upper and lower boundaries of an interstripe does not work in all cases. In particular, when a pattern is assigned $N_{cl} < 3$ from Step 4 this means that at least one “interstripe cluster” is, in fact, two interstripes connected by a bridge across a broken black stripe. Our approach does not extract boundaries that make intuitive sense in this case. Thus, in our results in section 4, we present summary statistics on interstripe–stripe boundary curviness for unbroken patterns. It would be interesting in the future to build on our methods to define and extract the upper and lower boundaries of interstripes when interruptions are present in the pattern. We suggest that the single and visible hyper-parameter α associated with constructing α -shapes makes them a nice tool to use when quantifying curviness, and we sweep across different values of α_m for $m \in \{\text{ABM}, \text{CA}\}$ to understand its impact in subsection 4.1.

4. Results: Quantitative study of on- and off-lattice microscopic models. We now apply our methods from section 3 to quantify stripe patterns resulting from two biologically detailed models of zebrafish, the ABM [102] and CA [80]. We begin by sweeping through different choices of the hyper-parameters in Figure 7 to determine their effects on pattern quantification (subsection 4.1). We then select values for our hyper-parameters that allow us to put simulated patterns from the models [80, 102] on equal footing and present a large-scale, discerning study of these models in subsection 4.2. See the caption of Figure 7 for a summary of the baseline hyper-parameter values that we establish in subsection 4.1 and use in subsection 4.2.

4.1. Robustness study: Choosing hyper-parameters. Our methods in Figure 7 depend on the choice of values for eight main hyper-parameters: four hyper-parameters associated with our initial data cleaning in Step 2, two hyper-parameters that emerge in Step 3 when we interpret persistent homology, one hyper-parameter related to clustering cells into interstripes in Step 4, and a final hyper-parameter that controls how tightly we estimate interstripe boundaries in Step 5. Importantly, it is not just choices of computational implementation, but also choices in methods for quantification, that affect how we interpret the output of different models. Because of this, here we sweep across values of our hyper-parameters to better understand their roles and select values that are biologically meaningful and robust. We focus on the four hyper-parameters involved in implementing and interpreting persistent homology and constructing α -shapes in Steps 3–5. (See section 5 for a discussion of the other hyper-parameters and additional choices in our pipeline.)

To interpret birth and death times of dimension-1 topological features as information about (inter)stripe width and number in Step 3 [68], we rely on a birth-time cutoff factor B_m and persistence threshold P_m , where $m \in \{\text{ABM}, \text{CA}\}$ denotes the potential for different values for the two models [80, 102]. For values $B_m = 1, 1.25, 1.5, \dots, 5.5, 5.75, 6$ and $P_m = 100, 120, 140, \dots, 460, 480, 500 \mu\text{m}$ in (3.1)–(3.3), we quantify 100 ABM and CA patterns, cropped and cleaned as described in subsection 3.1 and subsection 3.2, under different choices of B_m and P_m . In Figure 11, we show the effects of these hyper-parameters on the percentage of patterns that our methods identify as having broken stripes or interstripes. (We find that stripe and interstripe width is less sensitive to B_m and P_m ; see Supplementary Material (M154308_01.pdf [local/web 1.04MB]).) The extreme choice of $B_m = 1$ requires that loops corresponding to (inter)stripes be born at a scale strictly less than the mean distance between cells. This intuitively leads to no patterns being identified as having three interstripes

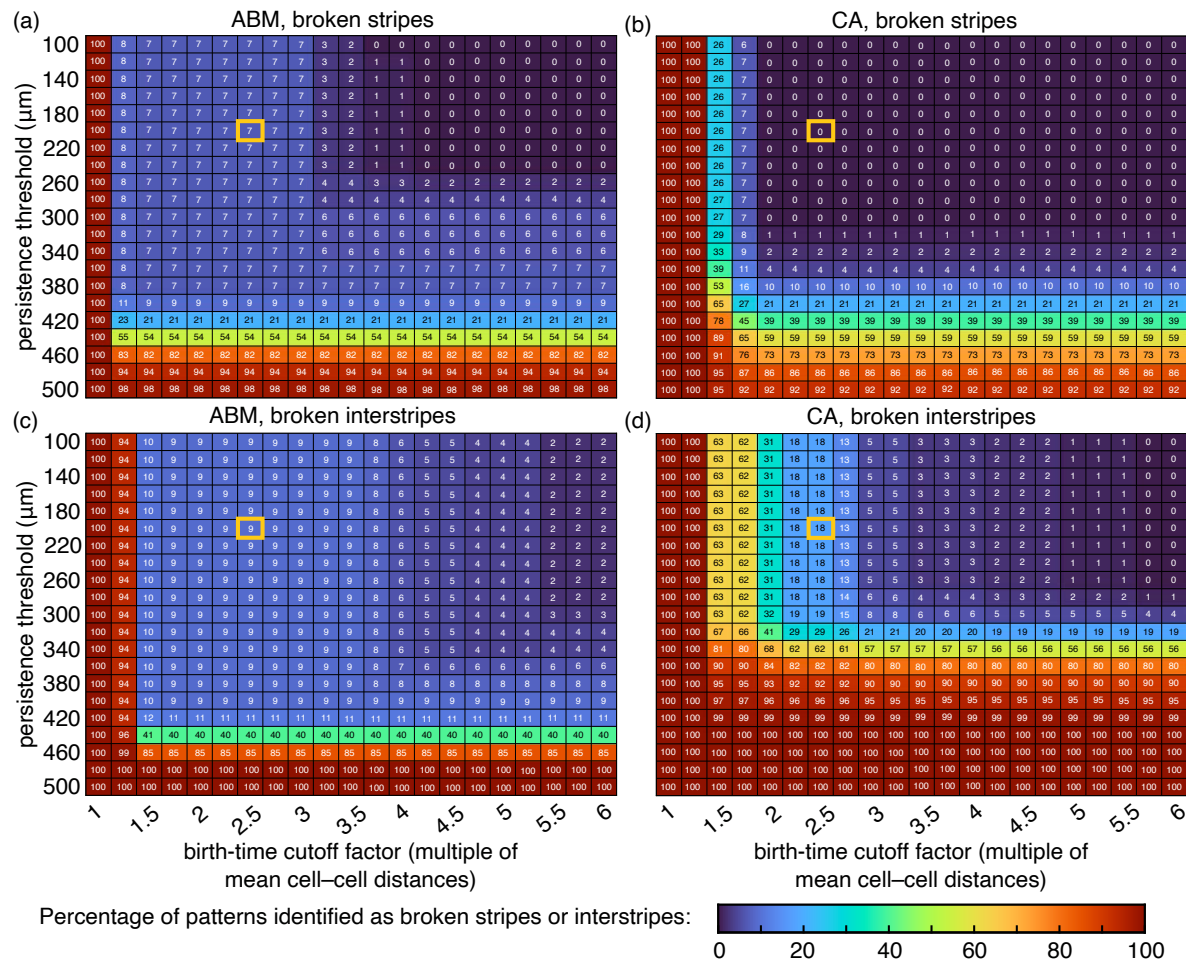


Figure 11. Effects of the birth-time cutoff factor B_m and persistence threshold P_m on identifying broken patterns in Step 3 (equations (3.1)–(3.3)). Results are based on 100 ABM [102] and 100 CA [80] patterns, cropped and cleaned as described in subsection 3.1 and subsection 3.2, that we quantify under each choice of (B_m, P_m) given. Here B_m is a factor multiplied by the appropriate mean cell–cell distance (see Figure 4(b)). (a) The number of ABM patterns flagged for broken stripes decreases as we relax how early loops must be born to count as stripes. (b) Our results for CA patterns show similar overall dependence on B_m and P_m , and we note that broken stripes are rare in CA patterns. (c) For large values of P_m , there are not sufficient topological loops of X^d cells with persistence P_m , so the number of ABM patterns flagged as having broken interstripes is 100%. (d) Our observations suggest that CA patterns feature some X^d cells that “infringe” on black stripe territory; this may be visible in 100% of CA patterns being flagged for broken interstripes at a lower value of P_m relative to ABM patterns. We highlight the baseline hyper-parameter values that we use in subsection 4.2 in yellow. The hyper-parameters B_m and P_m also play a role in how we quantify (inter)stripe width; see Supplementary Material (M154308.01.pdf [local/web 1.04MB]).

and two stripes; as expected, our methods flag 100% of patterns as broken when B_m is too low in Figure 11. Similarly, a persistence threshold P_m that is too large—e.g., beyond the width of interstripes—means our methods flag all patterns as broken because the number of dimension-1 topological features with large enough persistence is below our target of three.

Beyond low B_m and high P_m values, which are restrictive constraints on our inequalities in (3.1)–(3.3), we suggest that $B_m \geq 4$ is overly permissive of gaps between cells in (inter)stripes. This is based on empirical observations that zebrafish stripes are roughly 7–12 cells wide [75], so $B_m \geq 4$ means that stripes featuring gaps between adjacent cells comparable to half a stripe width pass through our methods without being flagged as broken. If we focus on $B_m < 4$ in Figure 11, there is a wide range of persistence thresholds under which our methods are insensitive. Choosing $P_{\text{ABM}} \in [100, 340]$ and $P_{\text{CA}} \in [100, 260]$ μm does not affect our results. (If we use different hyper-parameters for different cell types, we can consider a wider range for stripes, but we apply the same cutoff in (3.1)–(3.3).) In agreement with [68], we thus select $P_{\text{ABM}} = P_{\text{CA}} = 200$ μm as our baseline value. The sharp drop in ABM interstripes (Figure 11(c)) flagged as broken between $B_{\text{ABM}} = 1.25$ and 1.5 , followed by a fairly insensitive region, suggests that we should select $B_{\text{ABM}} \in [1.5, 3.75]$. For CA patterns, the number of interstripes flagged as broken (Figure 11(d)) repeatedly drops by about half as B_{CA} increases from 1.25 to 2.25 . In the end, choosing a value for B_m is a question of how large of a gap between adjacent cells we permit before calling an (inter)stripe broken, and we set $B_{\text{ABM}} = B_{\text{CA}} = 2.5$ cell-cell distances as our baseline value.

After using our baseline values of B_m and P_m to output the number of stripes and interstripes in Step 3, we turn to the hyper-parameters τ_m^{\min} and α_m that affect our measurements of curviness. We evaluate 100 CA and ABM patterns, excluding those with broken (inter)stripes, under minimum X^d cluster size $\tau_m^{\min} = 20, 30, \dots, 90, 100$ cells and α -radius $\alpha_m = 50, 60, \dots, 190, 200$ μm . Because McGuirl et al. [68] clustered cells into interstripes for ABM [102] patterns without any need to remove non-interstripe clusters first, we note that $\tau_{\text{ABM}}^{\min} = 0$ cells is valid. For CA patterns, as we show in Figures 12(a)–(b), boundary roughness makes it challenging to visually specify which X^d cells should be removed before computing curviness. Sweeping across values of τ_{CA}^{\min} and α_{CA} , all values of the minimum cluster size τ_m^{\min} between 20 and 100 cells lead to the same curviness results, so we do not show this heatmap. In contrast, choosing τ_{CA}^{\min} much lower than 20 cells does not reliably group X^d cells into N_{cl} clusters that are interstripes, and it causes errors in our algorithm: when τ_{CA}^{\min} is too low, outlier cells in CA patterns are sometimes selected as one of the interstripe clusters in Step 4. We thus choose $\tau_{\text{CA}}^{\min} = 60$ cells as our baseline value, and set $\tau_{\text{ABM}}^{\min} = \tau_{\text{CA}}^{\min}$ for consistency.

As we show in Figure 12(c), our results on stripe–interstripe boundary curviness for unbroken patterns across 100 ABM and CA simulations depend strongly on the choice of radius α_m that we use when constructing α -shapes in Step 5. This is particularly the case for CA patterns, which feature jagged interstripe edges. Selecting α_m means balancing sharply tracing out interstripe boundaries with relaxing our curves toward the convex hull of X^d positions. To gain intuition into the effects of α_m , we show the interstripe boundaries that we extracted and associated boundary curviness under different choices of α_m for two example ABM patterns in Figures 12(d)–(e) and two example CA patterns in Figures 12(f)–(g). The hyper-parameter α_m essentially controls how “forgiving” our methods are to peaks and crevices in X^d organization. Because the boundary curves for $\alpha_{\text{CA}} = 60$ μm are very rough and convoluted, we suggest that this value focuses on details that are too fine in CA patterns. On the other hand, selecting $\alpha_{\text{CA}} = 140$ μm or 180 μm in Figures 12(f)–(g) leads to boundary curves that miss structure and undulations in X^d organization that we suggest are meaningful. Supported by

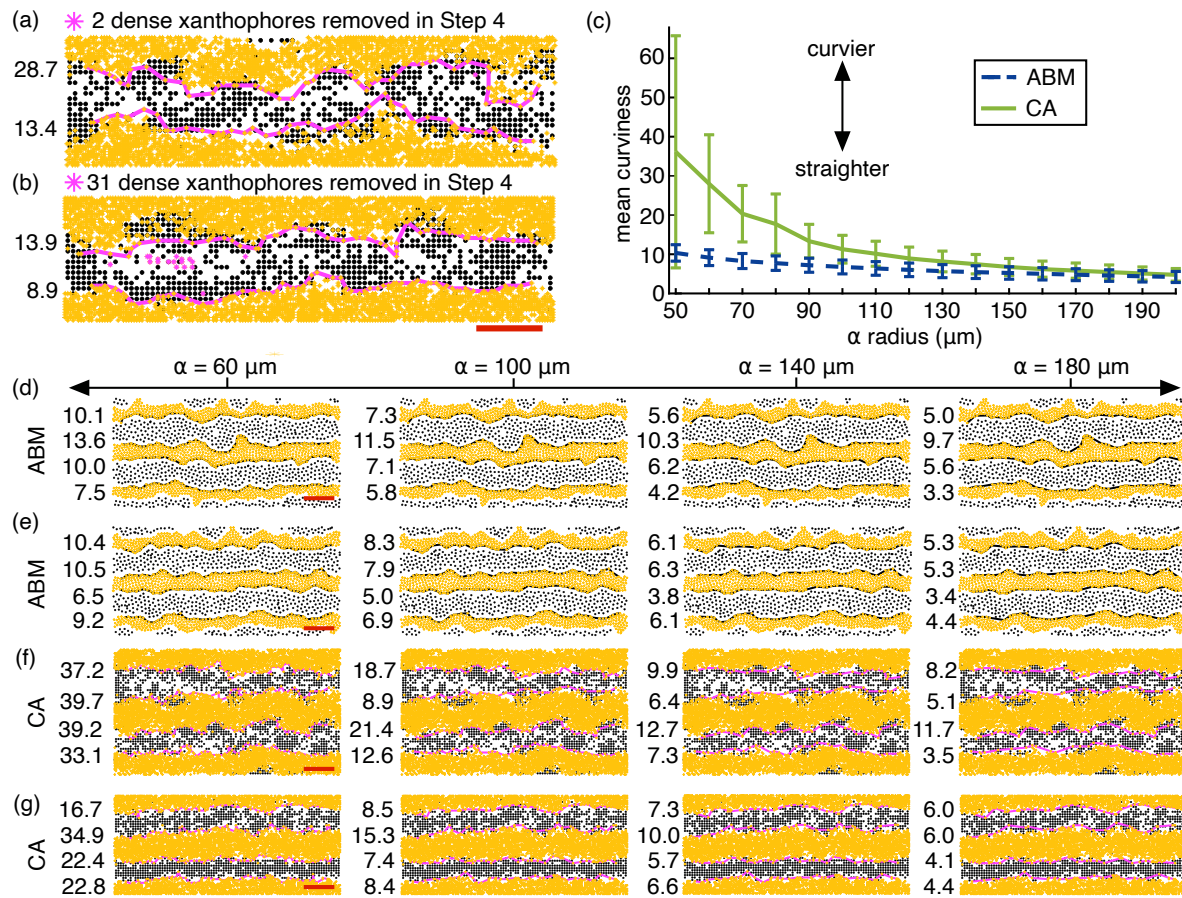


Figure 12. Effects of the minimum cluster size τ_m^{\min} and α -radius α_m on stripe–interstripe curviness in Steps 4–5. Our results are for 100 ABM [102] and CA [80] patterns, cropped and cleaned in Steps 1–2. We show summary statistics for the subset of these patterns classified as unbroken in Step 3 with the baseline hyper-parameter values established in Figure 11. (a)–(b) Some CA patterns [80] require further cleaning (i.e., removal of stray X^d cells and spots) to help ensure the N_{cl} groups that we cluster X^d cells into in Step 4 correspond to interstripes. We show a closer look at an example CA pattern with (a) two X^d cells removed before computing curviness, and one with (a) a larger spot removed. Removed cells are shown in magenta. Once τ_{CA}^{\min} is large enough to fall into the range $[20, 100]$ cells, our methods are insensitive to this choice. Based on [68], $\tau_{ABM}^{\min} = 0$ cells (i.e., no additional cleaning) is sufficient for ABM patterns. Our choice of baseline value $\tau_m^{\min} = 60$ cells for $m \in \{\text{ABM}, \text{CA}\}$ means that X^d clusters must contain at least 60 cells to be considered interstripes, and smaller clusters are removed. (c) As the radius α_m used to compute α -shapes grows, interstripe boundaries get closer to the convex hull of our X^d data. (d)–(e) We show two example ABM patterns with interstripe boundaries (black) extracted under different choices of α_{ABM} and (f)–(g) two example CA patterns with interstripe boundaries (magenta) associated with different α_{CA} values. The numbers to the left denote curviness; see (3.6). We use $\alpha_m = 100 \mu\text{m}$ for $m \in \{\text{ABM}, \text{CA}\}$ as our baseline value in subsection 4.2.

these observations and the fact that our curviness measurements for CA patterns show much less variance and change less rapidly as a function of α_{CA} after about $90 \mu\text{m}$ in Figure 12(c), we select $\alpha_{CA} = 100 \mu\text{m}$ as our baseline value. Again for consistency, we set $\alpha_{ABM} = 100 \mu\text{m}$.

4.2. Quantitative description of microscopic models at large scale. With the baseline values of our hyper-parameters from subsection 4.1 (see the caption of Figure 7 for a summary), we now present a large-scale study of 1000 ABM [102] and 1000 CA [80] patterns. As we show in Figure 13, our methods provide information on (inter)stripe width, stripe–interstripe boundary curviness, and the fraction of patterns with overall imperfections (e.g., broken stripes). We also track the number of non-interstripe X^d clusters removed in Step 4

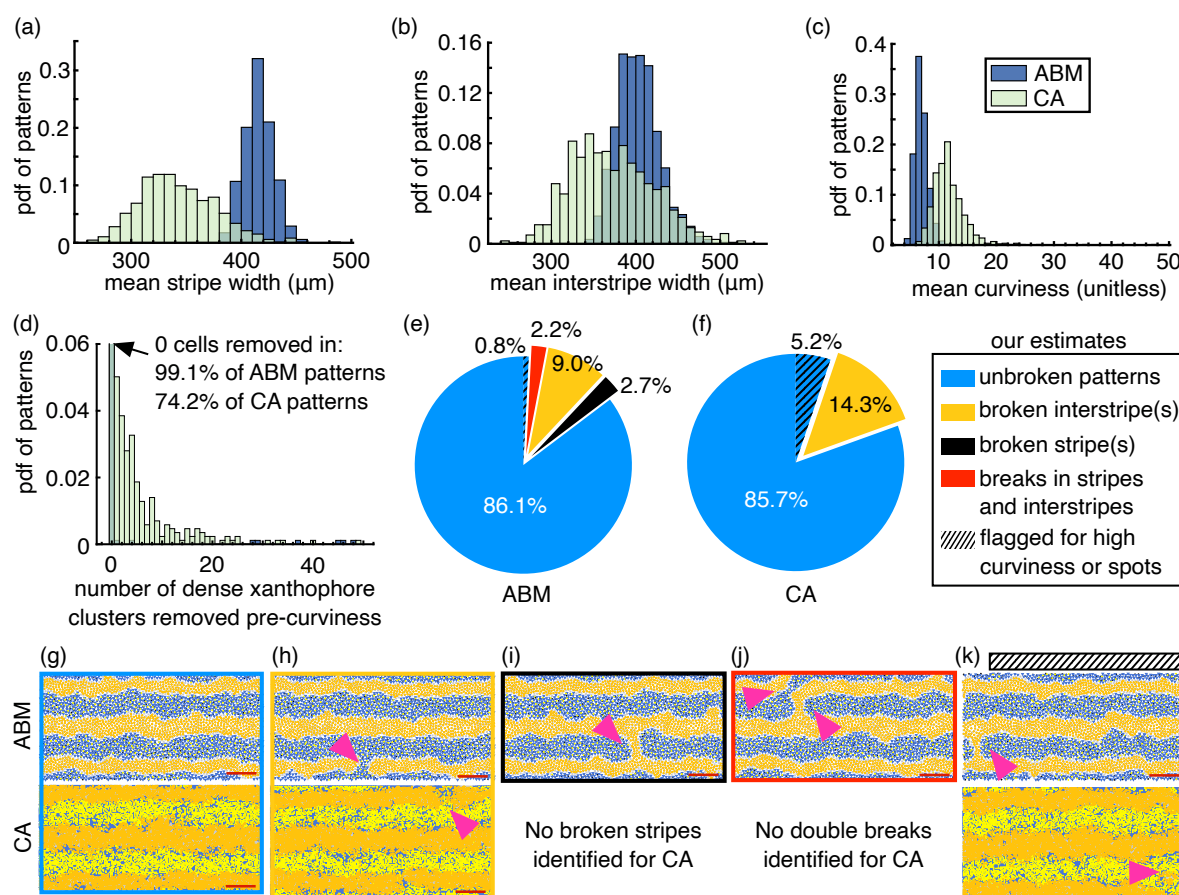


Figure 13. Baseline quantification of 1000 patterns generated from the ABM [102] and CA [80]. Distributions in (a)–(d) are for patterns identified as unbroken (namely, patterns with three interstripes and two stripes). (a) The overlap of stripe and interstripe cells in CA [80] patterns may play a role in why our methods estimate stripe width as narrower in CA patterns than in ABM [102] patterns. (b) With respect to interstripe width and other pattern features, CA [80] patterns are more variable than ABM [102] patterns. (c) High curviness may be associated with the presence of spots in patterns or with incorrectly classified broken patterns; see Figure 9. (d) To help account for spots in CA patterns, we further clean the data in Step 4; here we show the number of X^d clusters removed before computing curviness. (e) ABM [102] patterns are susceptible to breaks in stripes and interstripes, while (f) CA [80] are most susceptible to breaks in interstripes. For the ABM (top row) and CA (bottom row), we show example patterns that our methods identify (g) as unbroken, (h) as having broken interstripes, (i) as having unbroken stripes, or (j) as having breaks in stripes and interstripes. (k) While our methods classify these patterns as unbroken, they feature imperfections and represent false negatives in the form of a narrow break (ABM) and prominent spot (CA). Red scale bar in (g)–(k) is 500 μm .

before computing boundary curviness. Interpreting our results depends on viewpoint: from one perspective, the ABM and CA patterns are remarkably similar, particularly given the differences in noise structure and outlier cells that we accounted for in subsection 3.2. From another perspective, the results in Figure 13 show that the ABM and CA—despite being biologically similar—do produce different behaviors, most notably related to variability across simulations. Our quantitative approach allows us to move beyond viewing a few sample patterns from the stochastic models [80, 102] to better understand the features that these models give rise to at large scale.

As we show in Figure 13(a), the mean stripe width (across 1000 simulations) for the ABM [102] is about $416\ \mu\text{m}$, and the mean stripe width for the CA [80] is about $343\ \mu\text{m}$, a difference that corresponds to less than two M cells. The mean interstripe widths for the ABM and CA are closer: $403\ \mu\text{m}$ and $373\ \mu\text{m}$, respectively (see Figure 13(b)). This is a difference of less than one X^d cell from the perspective of the ABM [102] and less than two X^d cells from the perspective of the CA [80]. In terms of stripe–interstripe boundary curviness based on (3.6) (Figure 13(c)), the mean curviness of ABM patterns not flagged as broken is about 7.0 with a standard deviation of about 1.8. Similarly, the mean curviness of CA patterns is about 11.8 ± 3.0 . As we show in Figure 13(d), across the portion of our 1000 ABM and CA simulations that are not flagged as broken, 99.1% of ABM patterns do not undergo additional cleaning of X^d cells in Step 4. In comparison about a quarter of unbroken CA patterns contain small X^d clusters that we remove before computing interstripe–stripe boundary curviness.

At the pattern scale, we show the percentage of ABM and CA patterns that our methods identify as unbroken, as having broken interstripes only, as having broken stripes only, or as having breaks in stripes and interstripes in Figures 13(e)–(f). We find that our quantification pipeline reports a remarkably similar percentage of ABM [102] and CA [80] patterns as unbroken. Beyond this, ABM patterns [102] display breaks of different types, whereas CA patterns [100] are most susceptible to breaks in interstripes. Additionally, we flag—as a subset of patterns classified as “unbroken”—a rough estimate of the ABM and CA patterns that we suggest are false negatives. (False negatives are simulations that our methods should have classified as having pattern errors but did not.)

How we flag patterns as potential false negatives is model specific. In particular, based on our observations and the results [68], we expect dependable separation of X^d cells into interstripes in ABM [102] patterns. Thus, we suspect that the eight out of 1000 ABM patterns that are classified as unbroken, yet led to removal of some positive number of X^d cells in Step 4, are, in fact, broken. As we show in Figures 14(c)–(d) for two examples among these eight patterns, this is indeed the case. The eight ABM false negatives occur because B_{ABM} is just large enough so that the number of stripes from the perspective of either X^ℓ or M cells in (3.1) or (3.3) is two, allowing the narrow gaps in Figures 14(c)–(d) to pass by undetected. As a result, interstripes that are actually connected by a bridge are forced into three clusters, and the result is haphazard removal of X^d cells in Step 4.

On the other hand, we flag 52 “unbroken” CA patterns as potential false negatives in Figure 13(f). These fall into two groups: (1) patterns that require more than 25 X^d clusters to be removed in Step 4 before computing curviness, and (2) patterns with at least one stripe–interstripe boundary with curviness greater than 25. Patterns meeting these conditions are rare, as indicated by the distributions in Figures 13(c)–(d). We show a few example CA

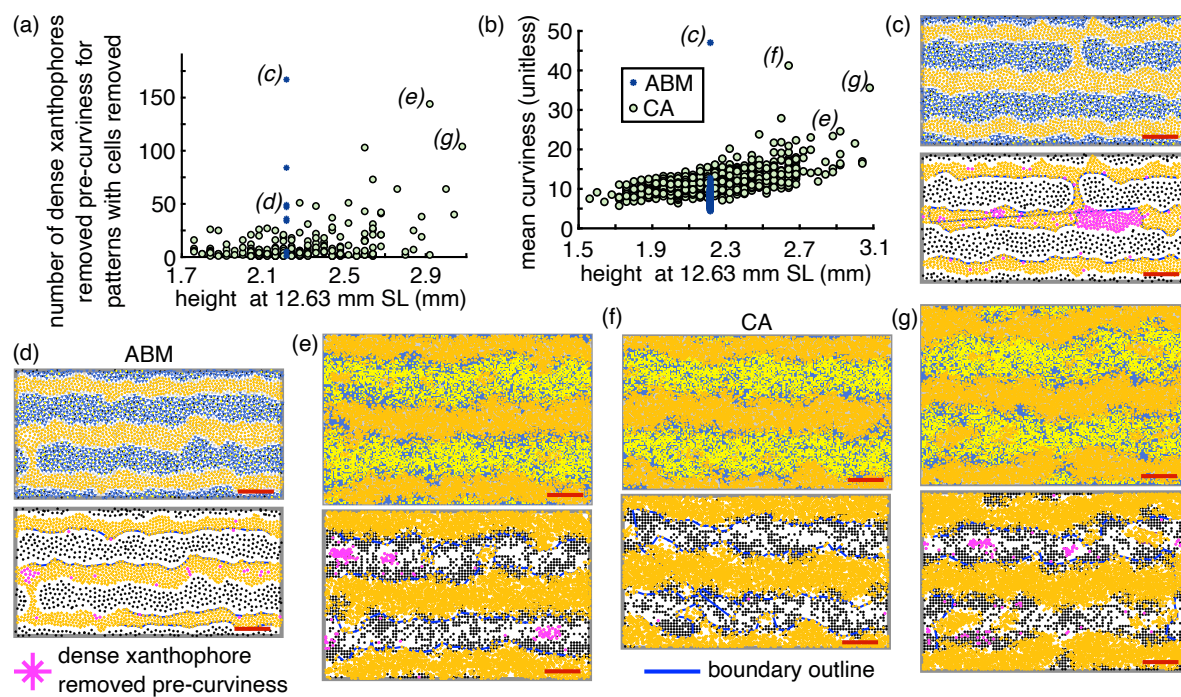


Figure 14. False negatives, spots, and the role of domain growth. (a) Across the subset of 1000 ABM and CA patterns identified by our methods as unbroken, we show the total number of X^d cells removed in Step 4 versus the domain height. Because the ABM [102] has deterministic domain growth, all scatter points are stacked at the same height; there are only eight patterns with more than zero X^d cells cleaned in Step 4. The CA [80] includes stochastic domain growth, meaning that domains of the same length have different heights. Interestingly, the spread of scatter points spreads with tall CA domains. (b) Removal of X^d cells in Step 4 and our curviness computations in Step 5 are related, since interstripe boundaries must curve and bend to encompass any X^d cells remaining in the pattern. We observe that the mean curviness of CA patterns increases and spreads with increasing domain height, particularly after CA domains become taller than ABM domains. The letters in (a), (b) refer to examples in (c)–(g). (c), (d) We show two example “unbroken” ABM [102] patterns; the top row provides the full pattern, and the bottom row focuses on M and X^d cells for clarity, with X^d cells removed in Step 4 indicated in magenta and interstripe boundaries shown in blue. We see odd behavior because X^d cells are being forced into three clusters, when only two are present. A narrow bridge is clearly visible in these patterns, and we conclude that all eight ABM [102] patterns with any removal of X^d cells in Step 4 are false negatives. (e)–(g) CA patterns with large numbers of X^d cells removed in Step 4 and high curviness often feature X^d spots. Comparing (f) and (g), for example, it can be challenging to judge what constitutes a spot versus a finger of an interstripe that should be included in its boundary. We flag CA [80] with very high curviness or a large number of X^d clusters removed in Step 4 as potentially spotted; see Figure 13.

patterns among these 52 potential false negatives in Figures 14(e)–(g): these patterns clearly include large X^d spots or tortuous boundaries. Because our methods do not directly quantify spots, flagging CA patterns in Figure 13(f) with very high curviness or many outlier X^d cells serves as a rough proxy for indicating that a subset of “unbroken” patterns feature spot imperfections. Adding to our methods and those in [68] to identify and characterize spots in CA patterns directly is an important direction for future research.

To further check our methodology, we viewed 200 patterns for each model. For the ABM [102], 171 of these patterns (85.5%) were classified as unbroken by our methods, and our qualitative observations agree in 170 cases. The single false negative features a narrow gap in a black stripe, and, as discussed above, it is flagged as potentially broken because it leads to more than zero X^d cells removed in Step 4. We agree with the 18 (9%), 8 (4%), and 3 (1.5%) patterns identified by our methods as having breaks in interstripes, stripes, or both, respectively. We had more difficulty classifying the CA [80] simulations visually, since these patterns feature more outliers and cells grouped less distinctly into stripes and interstripes. Across 200 CA patterns, our methods identified 0 with broken stripes (we agree) and 34 (17%) with broken interstripes (we suggest 7 of these are false positives and not broken). Among the 166 (83%) patterns that our methods identified as unbroken, we suggest four of these have interstripe breaks and three feature one or more prominent spots. Ten more of these unbroken patterns were flagged for high curviness as discussed above, and they all have at least one large visual spot. Our visual checks add confidence that our methods are capturing a meaningful overall picture of the patterns generated by both models [80, 102].

As a final note, we comment on a few interesting features of our curviness results in Figure 13(c). First, because X^d cells are more dense in CA patterns than in ABM patterns (see Figure 3), we thought this might lead our methods to interpret CA patterns as curvier than ABM patterns. To better understand this, we calculated the mean distance between pairs of X^d cells that share an edge in each graph representing an interstripe boundary, across all unbroken patterns. We found that the mean edge length in ABM boundary curves is about $62.7 \mu\text{m}$ —a little under twice the mean distance between neighboring X^d cells—and the mean edge length in CA boundary curves is about $83.8 \mu\text{m}$ —just under four times the mean distance between nearest X^d cells. This adds support to the conclusion that the difference in curviness between ABM [102] and CA [80] patterns is an inherent feature of the models. Moreover, we find that high curviness and high numbers of X^d clusters removed in Step 4 are more common in CA patterns on tall domains; see Figures 14(a)–(b). As we discuss in section 5, this has biological implications and may be related naturally to the interplay between nonlocal cell interactions and increased vertical domain growth.

5. Discussion and conclusions. Different modeling studies offer complementary perspectives and expertise on biological systems. However, as models become more detailed, it is more challenging to understand their behavior—let alone relate their dynamics to the behavior of other models. This is particularly true when stochastic realizations of the same model are variable. To help address this challenge in the case of zebrafish stripe patterns, we developed a method for quantitatively relating two stochastic, microscopic models of cell behavior at large scale. Patterns generated by the models [80, 102] in our case study are similar by limited observation, but feature important differences in cell density and outlier cells. Our approach uses tools from topological and geometric data analysis (namely persistent homology and α -shapes), as well as clustering and data cleaning, to count stripes and interstripes, estimate stripe width, and measure stripe–interstripe boundary curviness. We applied our methodology to the off-lattice model [102] and the on-lattice model [80]. These two models [80, 102], which are biologically similar but computationally and mathematically different, provide an excellent place to better understand the role of hyper-parameters in

our quantification pipeline. We swept across a range of values to choose hyper-parameters that allow for meaningful quantitative descriptions of the models [80, 102] alongside one another.

We found that ABM [102] and CA [80] patterns are susceptible to different types of noise and “imperfections.” (We use “imperfections” to broadly refer to patterns with broken stripes or interstripes, spots, or high curviness. Because large-scale quantitative data on *in vivo* zebrafish patterns is not available to the best of our knowledge, we do not know if these imperfections are biological errors.) ABM [102] patterns feature clear separation between stripe and interstripe cells, whereas the CA [80] often produces stray stripe cells in interstripes, and vice versa. We also observed differences in how ABM and CA patterns break: our methodology suggests that the ABM produces some patterns with breaks in stripes and interstripes. On the other hand, breaks in CA patterns are concentrated in interstripes. Across all of our quantification results, we found that CA [80] patterns are more variable than ABM [102] patterns. Interestingly, our methodology uncovers the presence of gold spots in some CA [80] stripes.

As we discussed in subsection 4.2, spots and highly curved interstripes are more common for CA simulations with above-average vertical domain growth. Since vertical growth stretches cell positions in a way that widens (inter)stripes, the appearance of spots on tall domains may be related to how nonlocal cell interactions in the models [80, 101, 102] constrain stripe width. In particular, based on empirical findings [75], M cells require signals from interstripe cells at long range for survival. Once stripe width is stretched too far by vertical domain growth in CA patterns, we would expect death of M cells in the middle of the interstripe, providing the opportunity for X^d cells to differentiate there. Thus, when vertical domain growth is high, we observe gold spots of X^d cells emerging in the middle of stripes like those in Figure 14(g). L-iridophores, an additional cell type that appears later in stripe formation and is not included in the models [80, 102], may play a role in pattern maintenance, helping prevent wide stripes from breaking into spots *in vivo* [37, 102]. As empirical data becomes available, better understanding the interplay of domain growth and pattern formation is an exciting direction for future work.

Our study focused on the role of hyper-parameters in interpreting persistent homology and constructing α -shapes. We did not sweep across the hyper-parameters involved in Steps 1 and 2 in part because computing persistent homology is the main computational bottleneck in our approach. We thus comment on the other hyper-parameters in our quantification pipeline and discuss some alternative choices here. First, cropping ABM and CA domains to the same length and removing the top 10% and bottom 10% of cells vertically in Step 1 may have affected our results. It is possible that this could cause some CA patterns with very curvy—but unbroken—(inter)stripes to appear broken on the cropped domains. It is thus worth keeping in mind that our results are descriptions of patterns in a focus domain region.

Second, we did not perform a robustness study of the hyper-parameters related to data cleaning in Step 2. For the ABM [102], as we noted in subsection 3.2, McGuirl et al. [68] showed that no cleaning is needed before computing persistent homology, so any reasonable hyper-parameter values (including removing no cells) in Step 2 works. For CA [80] patterns, data cleaning is much more important, and selecting the values of hyper-parameters in Step 2 is a balance between (1) preserving the structure of CA patterns and (2) removing outlier

cells that disrupt our interpretation of persistent homology. The choice of hyper-parameters that we made in subsection 3.2 allowed us to interpret topological summaries as information about stripes and interstripes in Step 3. Working with outliers when computing persistent homology is challenging in general, and it would be interesting to apply methods from statistical topological data analysis [4, 21, 23, 35] to zebrafish patterns in the future.

Third, we made choices about which cell types to base measurements on throughout our quantification process. For example, when characterizing the curviness of stripe–interstripe boundaries, we could have based our measurements on M or X^ℓ cells (i.e., curviness of black stripe boundaries), X^d cells (i.e., curviness of gold interstripe boundaries), or a combination of cells of several types. Like McGuirl et al. [68], we chose to focus on X^d cells. The overlap of M and X^d cells at stripe–interstripe boundaries in CA [80] patterns makes it challenging not only to visually trace out boundaries in CA patterns, but also to define (inter)stripe width. When looking at a pattern like the one in Figure 12(a), where does the top interstripe end and the stripe begin? What cells are part of interstripes, and what cells—if any—are present in spots? Indeed, when observing large sets of simulated patterns from the ABM [102] and particularly the CA [80], we found it very difficult to consistently judge model output visually. This further highlights the value of our automated approach and careful consideration of hyper-parameters when quantifying messy, cell-based patterns at large scale.

Both models [80, 102] simulate zebrafish mutant patterns, and McGuirl et al. [68] developed methods for quantifying several of these mutants for the ABM [102]. Future work could build on this approach [68] and our work to quantify CA spot patterns. In terms of (inter)stripe width and curviness, our work with stripe patterns shows that the on-lattice model [80] produces more variability across simulations than does the off-lattice model [102]. It would be interesting to investigate whether or not this is also the case for mutant patterns, and to identify what features of the models give rise to this difference in variability. In particular, throughout our work, we did not attempt to determine what causes the differences and similarities that our methods uncover between ABM [102] and CA [80] patterns. It may be the structure of stochasticity in cell behaviors, the parameters in model rules, choices of computational implementation, or something else that leads to these features. By developing quantitative methods that can be applied flexibly across models, our work helps opens the door to future studies that address mathematical questions like these and bring the perspectives of different microscopic models together to better understand biological systems.

Appendix A. Additional details on the two models in our case study. To give more intuition into our focal models [80, 102], we discuss their rules for two cell dynamics (as examples among many) in more detail here. First, the ABM [102] and CA [80] specify that one of the ways that the loose iridophore at position \mathbf{I}_j^ℓ can swap to dense is when two inequalities are met, namely,

$$(A.1) \quad \underbrace{\sum_{i=1}^{N_M} \mathbf{1}_{\Omega_{\text{loc}}^{\mathbf{I}_j^\ell}}(\mathbf{M}_i) < a \text{ and } \sum_{i=1}^{N_X^d} \mathbf{1}_{\Omega_{\text{long}}^{\mathbf{I}_j^\ell}}(\mathbf{X}_i^d) < b}_{\text{two conditions must be met}} \implies \mathbf{I}_j^\ell \text{ transforms to dense,}$$

where $a = 3$ and $b = 9$ cells for the ABM [102], and $a = 1$ and $b = 16$ cells for the CA [80]. Here N_X^d and N_M are the current numbers of X^d and M cells, respectively. The indicator functions

$\mathbf{1}_{\Omega_{\text{long}}^z}(\cdot)$ and $\mathbf{1}_{\Omega_{\text{loc}}^z}(\cdot)$ count the number of cells in a long-range annulus or short-range ball around \mathbf{z} ; see [80, 102] for length scales. In the ABM, \mathbf{I}_j^ℓ refers to the coordinates of the j th loose iridophore in continuous space, and, in the CA, the coordinates are in discrete space. Similarly, \mathbf{M}_i and \mathbf{X}_i^d mark the positions of the i th melanophore and dense xanthophore, respectively.

As an example of a dynamic that the models describe in a subtly different way, both models [80, 102] account for M differentiation from uniformly distributed precursors. Given a randomly selected location \mathbf{z} in the domain (either (x, y) -coordinates or a grid site), the ABM and CA treat \mathbf{z} as the location of a precursor that differentiates into an M cell according to long- and short-range signals. The ABM implements the rule

$$(A.2) \quad \underbrace{\sum_{i=1}^{N_X^d} \mathbf{1}_{\Omega_{\text{long}}^z}(\mathbf{X}_i^d) + \sum_{i=1}^{N_I^d} \mathbf{1}_{\Omega_{\text{long}}^z}(\mathbf{I}_i^d)}_{\text{long-range signals}} > \alpha + \beta \sum_{i=1}^{N_M} \mathbf{1}_{\Omega_{\text{long}}^z}(\mathbf{M}_i) \quad \text{and} \\ \underbrace{\sum_{i=1}^{N_X^d} \mathbf{1}_{\Omega_{\text{loc}}^z}(\mathbf{X}_i^d) + \sum_{i=1}^{N_M} \mathbf{1}_{\Omega_{\text{loc}}^z}(\mathbf{M}_i) + \sum_{i=1}^{N_I^d} \mathbf{1}_{\Omega_{\text{loc}}^z}(\mathbf{I}_i^d)}_{\text{local signals: competition and preventing overcrowding}} \leq \eta \implies M \text{ differentiates at } \mathbf{z},$$

where $\alpha = 3$ cells [101], $\beta = 3.5$, $\eta = 4$ cells, and N_I^d is the number of I^d cells. Using the same notation, the corresponding CA rule for M differentiation at the randomly selected grid position \mathbf{z} is

$$(A.3) \quad \underbrace{\sum_{i=1}^{N_X^d} \mathbf{1}_{\Omega_{\text{long}}^z}(\mathbf{X}_i^d) + \sum_{i=1}^{N_I^d} \mathbf{1}_{\Omega_{\text{long}}^z}(\mathbf{I}_i^d)}_{\text{long-range signals (same as Eqn. (A.2))}} > \alpha + \beta \sum_{i=1}^{N_M} \mathbf{1}_{\Omega_{\text{long}}^z}(\mathbf{M}_i) \quad \text{and} \\ \underbrace{\sum_{i=1}^{N_X^d} \mathbf{1}_{\Omega_{\text{loc}}^z}(\mathbf{X}_i^d) \leq \gamma \sum_{i=1}^{N_M} \mathbf{1}_{\Omega_{\text{loc}}^z}(\mathbf{M}_i) \quad \text{and} \quad \sum_{i=1}^{N_I^d} \mathbf{1}_{\Omega_{\text{loc}}^z}(\mathbf{I}_i^d) \leq \kappa}_{\text{local signals: competition}} \implies M \text{ differentiates at } \mathbf{z},$$

where $\alpha = 3$ cells, $\beta = 10$, $\gamma = 12$, and $\kappa = 3$ cells. The difference in β for the ABM and CA long-range terms in Eqn. (A.2) and (A.3) may be related to the difference in M density in these models; α is a delay term, slowing how quickly M cells differentiate after the appearance of interstripe cells.

Code availability. The MATLAB and Python programs that we developed to quantify on-lattice and off-lattice stripe patterns are publicly available on GitLab [24]. We also include a sample simulation from each model on GitLab [24]. A large set of example simulations from the agent-based model [102] is available at [69], and code to generate simulated data from the cellular automaton model is publicly available at [79].

REFERENCES

- [1] E. J. AMÉZQUITA, M. Y. QUIGLEY, T. A. OPHELDERS, E. MUNCH, AND D. H. CHITWOOD, *The shape of things to come: Topological data analysis and biology, from molecules to organisms*, Dev. Dyn., 249 (2020), pp. 816–833.
- [2] E. ARIAS-CASTRO AND A. RODRÍGUEZ-CASAL, *On estimating the perimeter using the alpha-shape*, Ann. Inst. Henri Poincaré Probab. Stat., 53 (2017), pp. 1051–1068.
- [3] U. BAUER, *Ripser: Efficient computation of Vietoris-Rips persistence barcodes*, J. Appl. Comput. Topol., 5 (2021), pp. 391–423.
- [4] P. BENDICH, T. GALKOVSKIY, AND J. HARER, *Improving homology estimates with random walks*, Inverse Problems, 27 (2011), 124002.
- [5] P. BENDICH, J. S. MARRON, E. MILLER, A. PIELOCH, AND S. SKWERER, *Persistent homology analysis of brain artery trees*, Ann. Appl. Stat., 10 (2016), pp. 198–218.
- [6] A. J. BERNOFF AND C. M. TOPAZ, *A primer of swarm equilibria*, SIAM J. Appl. Dyn. Syst., 10 (2011), pp. 212–250, <https://doi.org/10.1137/100804504>.
- [7] D. BHASKAR, A. MANHART, J. MILZMAN, J. T. NARDINI, K. M. STOREY, C. M. TOPAZ, AND L. ZIEGELMEIER, *Analyzing collective motion with machine learning and topology*, Chaos, 29 (2019), 123125.
- [8] D. BHASKAR, W. Y. ZHANG, AND I. Y. WONG, *Topological data analysis of collective and individual epithelial cells using persistent homology of loops*, Soft Matter, 17 (2021), pp. 4653–4664.
- [9] G. B. BLANCHARD, A. G. FLETCHER, AND L. J. SCHUMACHER, *The devil is in the mesoscale: Mechanical and behavioural heterogeneity in collective cell movement*, Semin. Cell Dev. Biol., 93 (2019), pp. 46–54.
- [10] J. M. BLOOMFIELD, K. J. PAINTER, AND J. A. SHERRATT, *How does cellular contact affect differentiation mediated pattern formation?*, Bull. Math. Biol., 73 (2011), pp. 1529–1558.
- [11] L. L. BONILLA, A. CARPIO, AND C. TRENADO, *Tracking collective cell motion by topological data analysis*, PLOS Comput. Biol., 16 (2020), e1008407.
- [12] E. H. BUDI, L. B. PATTERSON, AND D. M. PARICHY, *Post-embryonic nerve-associated precursors to adult pigment cells: Genetic requirements and dynamics of morphogenesis and differentiation*, PLoS Genet., 7 (2011), e1002044.
- [13] D. BULLARA AND Y. DE DECKER, *Pigment cell movement is not required for generation of Turing patterns in zebrafish skin*, Nat. Commun., 6 (2015), 6971.
- [14] A. BUTTENSCHÖN AND L. EDELSTEIN-KESHET, *Bridging from single to collective cell migration: A review of models and links to experiments*, PLOS Comput. Biol., 16 (2020).
- [15] J. W. CAHN AND J. E. HILLIARD, *Free energy of a nonuniform system. I. Interfacial free energy*, Chem. Phys., 28 (1958), pp. 258–267.
- [16] C. E. CAICEDO-CARVAJAL AND T. SHINBROT, *In silico zebrafish pattern formation*, Dev. Biol., 315 (2008), pp. 397–403.
- [17] G. CARLSSON, *Topology and data*, Bull. Amer. Math. Soc., 46 (2009), pp. 255–308.
- [18] G. CARLSSON, *Topological methods for data modelling*, Nat. Rev. Phys., 2 (2020), pp. 697–708.
- [19] G. CARLSSON, A. ZOMORODIAN, A. COLLINS, AND L. GUIBAS, *Persistence barcodes for shapes*, Int. J. Shape Model., 11 (2005), pp. 149–187.
- [20] F. CHAZAL, V. DE SILVA, M. GLISSE, AND S. OUDOT, *The Structure and Stability of Persistence Modules*, Springer, 2016.
- [21] F. CHAZAL, B. FASY, F. LECCI, B. MICHEL, A. RINALDO, AND L. WASSERMAN, *Robust topological inference: Distance to a measure and kernel distance*, J. Mach. Learn. Res., 18 (2018), pp. 5854–5884.
- [22] Y.-L. CHUANG, M. R. D’ORSOGNA, D. MARTHALER, A. L. BERTOZZI, AND L. S. CHAYES, *State transitions and the continuum limit for a 2D interacting, self-propelled particle system*, Phys. D, 232 (2007), pp. 33–47.
- [23] M.-V. CIOCANEL, R. JUENEMANN, A. T. DAWES, AND S. A. MCKINLEY, *Topological data analysis approaches to uncovering the timing of ring structure onset in filamentous networks*, Bull. Math. Biol., 83 (2021), 21.

- [24] E. CLEVELAND, A. ZHU, B. SANDSTEDTE, AND A. VOLKENING, *Quantifying Different Models of Zebrafish Stripes Using TDA*, 2023, <https://gitlab.com/alexandriavolkening/quantifying-different-models-of-zebrafish-stripes-using-TDA>.
- [25] I. D. COUZIN, J. KRAUSE, R. JAMES, G. D. RUXTON, AND N. R. FRANKS, *Collective memory and spatial sorting in animal groups*, J. Theoret. Biol., 218 (2002), pp. 1–11.
- [26] S. DINI, B. J. BINDER, AND J. E. F. GREEN, *Understanding interactions between populations: Individual based modelling and quantification using pair correlation functions*, J. Theoret. Biol., 439 (2018), pp. 50–64.
- [27] C. M. DOOLEY, A. MONGERA, B. WALDERICH, AND C. NÜSSLEIN-VOLHARD, *On the embryonic origin of adult melanophores: The role of ErbB and Kit signalling in establishing melanophore stem cells in zebrafish*, Development, 140 (2013), pp. 1003–1013.
- [28] H. EDELSBRUNNER, *Alpha shapes—a survey*, in Tesselations in the Sciences, 2010.
- [29] H. EDELSBRUNNER AND J. HARER, *Persistent homology—a survey*, Contemp. Math., 453 (2008), pp. 257–282.
- [30] H. EDELSBRUNNER, D. KIRKPATRICK, AND R. SEIDEL, *On the shape of a set of points in the plane*, IEEE Trans. Inform. Theory., 29 (1983), pp. 551–559.
- [31] H. EDELSBRUNNER AND E. P. MÜCKE, *Three-dimensional alpha shapes*, ACM Trans. Graphics, 13 (1994), pp. 43–72.
- [32] D. S. EOM AND D. M. PARICHY, *A macrophage relay for long-distance signaling during postembryonic tissue remodeling*, Science, 355 (2017), pp. 1317–1320.
- [33] A. FADEEV, J. KRAUSS, H. G. FROHNHÖFER, U. IRION, AND C. NÜSSLEIN-VOLHARD, *Tight junction protein 1a regulates pigment cell organisation during zebrafish colour patterning*, eLife, 4 (2015), e06545.
- [34] A. FADEEV, J. KRAUSS, A. P. SINGH, AND C. NÜSSLEIN-VOLHARD, *Zebrafish leucocyte tyrosine kinase controls iridophore establishment, proliferation and survival*, Pigment Cell Melanoma Res., 29 (2016), pp. 284–296.
- [35] B. T. FASY, F. LECCI, A. RINALDO, L. WASSERMAN, S. BALAKRISHNAN, AND A. SINGH, *Confidence sets for persistence diagrams*, Ann. Statist., 42 (2014), pp. 2301–2339.
- [36] H. G. FROHNHÖFER, S. GEIGER-RUDOLPH, M. PATTKY, M. MEIXNER, C. HUHN, H.-M. MAISCHEIN, R. GEISLER, I. GEHRING, F. MADERSPACHER, C. NÜSSLEIN-VOLHARD, AND U. IRION, *Spermidine, but not spermine, is essential for pigment pattern formation in zebrafish*, Biol. Open, 5 (2016), pp. 736–744.
- [37] H. G. FROHNHÖFER, J. KRAUSS, H. M. MAISCHEIN, AND C. NÜSSLEIN-VOLHARD, *Iridophores and their interactions with other chromatophores are required for stripe formation in zebrafish*, Development, 140 (2013), pp. 2997–3007.
- [38] E. A. GAFFNEY AND S. SEIRIN LEE, *The sensitivity of Turing self-organization to biological feedback delays: 2D models of fish pigmentation*, Math. Med. Biol., 32 (2015), pp. 57–79.
- [39] J. D. GARDINER, J. BEHNSEN, AND C. A. BRASSEY, *Alpha shapes: Determining 3D shape complexity across morphologically diverse structures*, BMC Evol. Biol., 18 (2018), 184.
- [40] E. GAVAGNIN, J. P. OWEN, AND C. A. YATES, *Pair correlation functions for identifying spatial correlation in discrete domains*, Phys. Rev. E, 97 (2018), 062104.
- [41] R. GHRIST, *Elementary Applied Topology*, Createspace, 2014.
- [42] A. GIERER AND H. MEINHARDT, *A theory of biological pattern formation*, Kybernetik, 12 (1972), pp. 30–39.
- [43] R. GINIUNAITĖ, R. E. BAKER, P. M. KULESA, AND P. K. MAINI, *Modelling collective cell migration: Neural crest as a model paradigm*, J. Math. Biol., 80 (2020), pp. 481–504.
- [44] J. A. GLAZIER AND F. GRANER, *Simulation of the differential adhesion driven rearrangement of biological cells*, Phys. Rev. E, 47 (1993), pp. 2128–2154.
- [45] K. GOWDA, H. RIECKE, AND M. SILBER, *Transitions between patterned states in vegetation models for semiarid ecosystems*, Phys. Rev. E, 89 (2014), 022701.
- [46] B. GUO, J. MENON, AND B. WILLETTE, *Surface reconstruction using Alpha shapes*, Comput. Graphics Forum, 16 (1997), pp. 177–190.

- [47] D. GUR, E. J. BAIN, K. R. JOHNSON, A. J. AMAN, H. AMALIA PASOLLI, J. D. FLYNN, M. C. ALLEN, D. D. DEHEYN, J. C. LEE, J. LIPPINCOTT-SCHWARTZ, AND D. M. PARICHY, *In situ differentiation of iridophore crystallotypes underlies zebrafish stripe patterning*, Nat. Commun., 11 (2020), 6391.
- [48] H. HAMADA, M. WATANABE, H. E. LAU, T. NISHIDA, T. HASEGAWA, D. M. PARICHY, AND S. KONDO, *Involvement of Delta/Notch signaling in zebrafish adult pigment stripe patterning*, Development, 141 (2014), pp. 318–324.
- [49] T. HIRASHIMA, E. G. RENS, AND R. M. H. MERKS, *Cellular Potts modeling of complex multicellular behaviors in tissue morphogenesis*, Dev. Growth Differ., 59 (2017), pp. 329–339.
- [50] M. HIRATA, K.-I. NAKAMURA, T. KANEMARU, Y. SHIBATA, AND S. KONDO, *Pigment cell organization in the hypodermis of zebrafish*, Dev. Dyn., 227 (2003), pp. 497–503.
- [51] M. HIRATA, K.-I. NAKAMURA, AND S. KONDO, *Pigment cell distributions in different tissues of the zebrafish, with special reference to the striped pigment pattern*, Dev. Dyn., 234 (2005), pp. 293–300.
- [52] C. HUEPE AND M. ALDANA, *New tools for characterizing swarming systems: A comparison of minimal models*, Phys. A, 387 (2008), pp. 2809–2822.
- [53] M. INABA, H. YAMANAKA, AND S. KONDO, *Pigment pattern formation by contact-dependent depolarization*, Science, 335 (2012), pp. 677–677.
- [54] U. IRION, H. G. FROHNHÖFER, J. KRAUSS, T. C. CHAMPOLLION, H. MAISCHIN, S. GEIGER-RUDOLPH, C. WEILER, AND C. NÜSSLEIN-VOLHARD, *Gap junctions composed of connexins 41.8 and 39.4 are essential for colour pattern formation in zebrafish*, eLife, 3 (2014), e05125.
- [55] U. IRION AND C. NÜSSLEIN-VOLHARD, *The identification of genes involved in the evolution of color patterns in fish*, Curr. Opin. Genet. Dev., 57 (2019), pp. 31–38.
- [56] M. IWASHITA, M. WATANABE, M. ISHII, T. CHEN, S. L. JOHNSON, Y. KURACHI, N. OKADA, AND S. KONDO, *Pigment pattern in jaguar/obelix zebrafish is caused by a Kir7.1 mutation: Implications for the regulation of melanosome movement*, PLoS Genet., 2 (2006), e197.
- [57] Y. KATZ, K. TUNSTRØM, C. C. IOANNOU, C. HUEPE, AND I. D. COUZIN, *Inferring the structure and dynamics of interactions in schooling fish*, Proc. Natl. Acad. Sci. USA, 108 (2011), pp. 18720–18725.
- [58] S. KONDO, M. WATANABE, AND S. MIYAZAWA, *Studies of Turing pattern formation in zebrafish skin*, Philos. Trans. Roy. Soc. A, 379 (2021), 20200274.
- [59] C. KONOW, Z. LI, S. SHEPHERD, D. BULLARA, AND I. R. EPSTEIN, *Influence of survival, promotion, and growth on pattern formation in zebrafish skin*, Sci. Rep., 11 (2021), 9864.
- [60] J. KURSAWE, R. E. BAKER, AND A. G. FLETCHER, *Impact of implementation choices on quantitative predictions of cell-based computational models*, J. Comput. Phys., 345 (2017), pp. 752–767.
- [61] L. LI, S. RUI, Q. NIE, X. GONG, AND F. LI, *Conformal alpha shape-based multi-scale curvature estimation from point clouds*, J. Comput., 7 (2012), pp. 1460–1466.
- [62] J. LIANG, H. EDELSBRUNNER, P. FU, P. V. SUDHAKAR, AND S. SUBRAMANIAM, *Analytical shape computation of macromolecules: I. Molecular area and volume through alpha shape*, Proteins, 33 (1998), pp. 1–17.
- [63] Q.-X. LIU, A. DOELMAN, V. ROTTSCHAFER, M. DE JAGER, P. M. J. HERMAN, M. RIETKERK, AND J. VAN DE KOPPEL, *Phase separation explains a new class of self-organized spatial patterns in ecological systems*, Proc. Natl. Acad. Sci. USA, 110 (2013), pp. 11905–11910.
- [64] F. MADERSPACHER AND C. NÜSSLEIN-VOLHARD, *Formation of the adult pigment pattern in zebrafish requires leopard and obelix dependent cell interactions*, Development, 130 (2003), pp. 3447–3457.
- [65] P. MAHALWAR, A. P. SINGH, A. FADEEV, C. NÜSSLEIN-VOLHARD, AND U. IRION, *Heterotypic interactions regulate cell shape and density during color pattern formation in zebrafish*, Biol. Open, 5 (2016), pp. 1680–1690.
- [66] P. MAHALWAR, B. WALDERICH, A. P. SINGH, AND C. NÜSSLEIN-VOLHARD, *Local reorganization of xanthophores fine-tunes and colors the striped pattern of zebrafish*, Science, 345 (2014), pp. 1362–1364.
- [67] S. G. MCCALLA AND J. H. VON BRECHT, *Consistent dynamics of stripes formed by cell-type interfaces*, SIAM J. Appl. Dyn. Syst., 17 (2018), pp. 2615–2633, <https://doi.org/10.1137/18M1173320>.
- [68] M. R. MCGUIRL, A. VOLKENING, AND B. SANDSTEDE, *Topological data analysis of zebrafish patterns*, Proc. Natl. Acad. Sci. USA, 117 (2020), pp. 5113–5124.
- [69] M. R. MCGUIRL, A. VOLKENING, AND B. SANDSTEDE, *Zebrafish Simulation Data*, 2020, https://figshare.com/projects/Zebrafish_simulation_data/72689.

- [70] S. K. McMENAMIN, E. J. BAIN, A. E. McCANN, L. B. PATTERSON, D. S. EOM, Z. P. WALLER, J. C. HAMILL, J. A. KUHLMAN, J. S. EISEN, AND D. M. PARICHY, *Thyroid hormone-dependent adult pigment cell lineage and pattern in zebrafish*, *Science*, 345 (2014), pp. 1358–1361.
- [71] S. MIYAZAWA, *Pattern blending enriches the diversity of animal colorations*, *Sci. Adv.*, 6 (2020), eabb9107.
- [72] S. MIYAZAWA, M. OKAMOTO, AND S. KONDO, *Blending of animal colour patterns by hybridization*, *Nat. Commun.*, 1 (2010), 66.
- [73] J. MOREIRA AND A. DEUTSCH, *Pigment pattern formation in zebrafish during late larval stages: A model based on local interactions*, *Dev. Dyn.*, 232 (2005), pp. 33–42.
- [74] E. MUNCH, *A user's guide to topological data analysis*, *J. Learn. Anal.*, 4 (2017), pp. 47–61.
- [75] A. NAKAMASU, G. TAKAHASHI, A. KANBE, AND S. KONDO, *Interactions between zebrafish pigment cells responsible for the generation of Turing patterns*, *Proc. Natl. Acad. Sci. USA*, 106 (2009), pp. 8429–8434.
- [76] J. T. NARDINI, B. J. STOLZ, K. B. FLORES, H. A. HARRINGTON, AND H. M. BYRNE, *Topological data analysis distinguishes parameter regimes in the Anderson-Chaplain model of angiogenesis*, *PLOS Comput. Biol.*, 17 (2021), e1009094.
- [77] J. M. OSBORNE, A. G. FLETCHER, J. M. PITT-FRANCIS, P. K. MAINI, AND D. J. GAVAGHAN, *Comparing individual-based approaches to modelling the self-organization of multicellular tissues*, *PLOS Comput. Biol.*, 13 (2017), pp. 1–34.
- [78] N. OTTER, M. A. PORTER, U. TILLMANN, P. GRINDROD, AND H. A. HARRINGTON, *A roadmap for the computation of persistent homology*, *EPJ Data Sci.*, 6 (2017), pp. 1–38.
- [79] J. P. OWEN, R. N. KELSH, AND C. A. YATES, *Zebrafish-stripe-Model*, 2020, <https://github.com/JenniferOwen/Zebrafish-stripe-model>.
- [80] J. P. OWEN, R. N. KELSH, AND C. A. YATES, *A quantitative modelling approach to zebrafish pigment pattern formation*, *eLife*, 9 (2020), e52998.
- [81] K. J. PAINTER, J. M. BLOOMFIELD, J. A. SHERRATT, AND A. GERISCH, *A nonlocal model for contact attraction and repulsion in heterogeneous cell populations*, *Bull. Math. Biol.*, 77 (2015), pp. 1132–1165.
- [82] D. M. PARICHY, *Evolution of pigment cells and patterns: Recent insights from teleost fishes*, *Curr. Opin. Genet. Dev.*, 69 (2021), pp. 88–96.
- [83] D. M. PARICHY, M. R. ELIZONDO, M. G. MILLS, T. N. GORDON, AND R. E. ENGESZER, *Normal table of postembryonic zebrafish development: Staging by externally visible anatomy of the living fish*, *Dev. Dyn.*, 238 (2009), pp. 2975–3015.
- [84] D. M. PARICHY AND J. M. TURNER, *Temporal and cellular requirements for Fms signaling during zebrafish adult pigment pattern development*, *Development*, 130 (2003), pp. 817–833.
- [85] D. M. PARICHY AND J. M. TURNER, *Zebrafish puma mutant decouples pigment pattern and somatic metamorphosis*, *Dev. Biol.*, 256 (2003), pp. 242–257.
- [86] L. B. PATTERSON, E. J. BAIN, AND D. M. PARICHY, *Pigment cell interactions and differential xanthophore recruitment underlying zebrafish stripe reiteration and Danio pattern evolution*, *Nat. Commun.*, 5 (2014), 5299.
- [87] L. B. PATTERSON AND D. M. PARICHY, *Interactions with iridophores and the tissue environment required for patterning melanophores and xanthophores during zebrafish adult pigment stripe formation*, *PLOS Genet.*, 9 (2013), e1003561.
- [88] L. B. PATTERSON AND D. M. PARICHY, *Zebrafish pigment pattern formation: Insights into the development and evolution of adult form*, *Annu. Rev. Genet.*, 53 (2019), pp. 505–530.
- [89] M. J. PLANK AND M. J. SIMPSON, *Models of collective cell behaviour with crowding effects: Comparing lattice-based and lattice-free approaches*, *J. R. Soc. Interface*, 9 (2012), pp. 2983–2996.
- [90] A. P. SINGH AND C. NÜSSLEIN-VOLHARD, *Zebrafish stripes as a model for vertebrate colour pattern formation*, *Curr. Biol.*, 25 (2015), pp. R81–R92.
- [91] A. P. SINGH, U. SCHACH, AND C. NÜSSLEIN-VOLHARD, *Proliferation, dispersal and patterned aggregation of iridophores in the skin prefigure striped colouration of zebrafish*, *Nat. Cell Biol.*, 16 (2014), pp. 604–611.
- [92] B. J. STOLZ, J. KAEPLER, B. MARKELC, F. BRAUN, F. LIPSMEIER, R. J. MUSCHEL, H. M. BYRNE, AND H. A. HARRINGTON, *Multiscale topology characterizes dynamic tumor vascular networks*, *Sci. Adv.*, 8 (2022), eabm2456.

- [93] G. TAKAHASHI AND S. KONDO, *Melanophores in the stripes of adult zebrafish do not have the nature to gather, but disperse when they have the space to move*, *Pigment Cell Melanoma Res.*, 21 (2008), pp. 677–686.
- [94] C. M. TOPAZ, L. ZIEGELMEIER, AND T. HALVERSON, *Topological data analysis of biological aggregation models*, *PLOS ONE*, 10 (2015), e0126383.
- [95] K. K. TRELOAR, M. J. SIMPSON, B. J. BINDER, D. L. S. McELWAIN, AND R. E. BAKER, *Assessing the role of spatial correlations during collective cell spreading*, *Sci. Rep.*, 4 (2014), 5713.
- [96] A. M. TURING, *The chemical basis of morphogenesis*, *Philos. Trans. R. Soc. Lond. B Biol. Sci.*, 237 (1952), pp. 37–72.
- [97] M. ULMER, L. ZIEGELMEIER, AND C. M. TOPAZ, *A topological approach to selecting models of biological experiments*, *PLOS ONE*, 14 (2019), e0213679.
- [98] A. VOLKENING, *Modeling Pattern Formation on Zebrafish*, Ph.D. thesis, Brown University, 2017, <https://doi.org/10.7301/Z0F18X6B>.
- [99] A. VOLKENING, *Linking genotype, cell behavior, and phenotype: Multidisciplinary perspectives with a basis in zebrafish patterns*, *Curr. Opin. Genet. Dev.*, 63 (2020), pp. 78–85.
- [100] A. VOLKENING, M. R. ABBOTT, N. CHANDRA, B. DUBOIS, F. LIM, D. SEXTON, AND B. SANDSTEDE, *Modeling stripe formation on growing zebrafish tailfins*, *Bull. Math. Biol.*, 82 (2020), 56.
- [101] A. VOLKENING AND B. SANDSTEDE, *Modelling stripe formation in zebrafish: An agent-based approach*, *J. R. Soc. Interface.*, 12 (2015), 20150812.
- [102] A. VOLKENING AND B. SANDSTEDE, *Iridophores as a source of robustness in zebrafish stripes and variability in Danio patterns*, *Nat. Commun.*, 9 (2018), 3231.
- [103] M. WATANABE, M. IWASHITA, M. ISHII, Y. KURACHI, A. KAWAKAMI, S. KONDO, AND N. OKADA, *Spot pattern of leopard Danio is caused by mutation in the zebrafish connexin41.8 gene*, *EMBO Rep.*, 7 (2006), pp. 893–897.
- [104] M. WATANABE AND S. KONDO, *Is pigment patterning in fish skin determined by the Turing mechanism?*, *Trends Genet.*, 31 (2015), pp. 88–96.
- [105] T. E. WOOLLEY, *Pattern production through a chiral chasing mechanism*, *Phys. Rev. E*, 96 (2017), 032401.
- [106] T. E. WOOLLEY, P. K. MAINI, AND E. A. GAFFNEY, *Is pigment cell pattern formation in zebrafish a game of cops and robbers?*, *Pigment Cell Melanoma Res.*, 27 (2014), pp. 686–687.
- [107] X. XU AND K. HARADA, *Automatic surface reconstruction with alpha-shape method*, *Vis. Comput.*, 19 (2003), pp. 431–443.
- [108] M. YAMAGUCHI, E. YOSHIMOTO, AND S. KONDO, *Pattern regulation in the stripe of zebrafish suggests an underlying dynamic and autonomous mechanism*, *Proc. Natl. Acad. Sci. USA*, 104 (2007), pp. 4790–4793.
- [109] W. ZHOU AND H. YAN, *Alpha shape and Delaunay triangulation in studies of protein-related interactions*, *Brief Bioinform.*, 15 (2012), pp. 54–64.