The Design Principles of Discrete Turing Patterning Systems

Tom Leyshon[1], Elisa Tonello[2], David Schnoerr[1], Heike Siebert[2] & Michael P.H. Stumpf[1,3,†]

[1] Department of Life Sciences, Imperial College London, London SW7 2AZ, UK
[2] FB Mathematik und Informatik, Freie Universitt Berlin, D-14195 Berlin, Germany
[3] School of BioSciences and School of Mathematics and Statistics, University of Melbourne, Parkville VIC 3010, Australia
[†] mstumpf@unimelb.edu.au

Abstract

The formation of spatial structures lies at the heart of developmental processes. However, many of the underlying gene regulatory and biochemical processes remain poorly understood. Turing patterns constitute a main candidate to explain such processes, but they appear sensitive to fluctuations and variations in kinetic parameters, raising the question of how they may be adopted and realised in naturally evolved systems. The vast majority of mathematical studies of Turing patterns have used continuous models specified in terms of partial differential equations. Here, we complement this work by studying Turing patterns using discrete cellular automata models. We perform a large-scale study on all possible two-node networks and find the same Turing pattern producing networks as in the continuous framework. In contrast to continuous models, however, we find the Turing topologies to be substantially more robust to changes in the parameters of the model. We also find that Turing instabilities are a much weaker predictor for emerging patterns in simulations in our discrete modelling framework. We propose a modification of the definition of a Turing instability for cellular automata models as a better predictor. The similarity of the results for the two modelling frameworks suggests a deeper underlying principle of Turing mechanisms in nature. Together with the larger robustness in the discrete case this suggests that Turing patterns may be more robust than previously thought.

1 Introduction

Nature is full of highly structured multi-cellular organisms that develop from single fertilized cells. It remains a key question how these can evolve robustly in the presence of environmental fluctuations.

The Turing mechanism has been proposed to explain such developmental patterning processes. It was first proposed by Alan Turing in 1952 [1]. The Turing mechanism gives rise to self-organised patterns in the local concentrations of biochemical components in reaction-diffusion systems. These lead to patterns such as spots, stripes and labyrinths [2]. Such inhomogenous patterns are induced by diffusion of the components. Due to this counter-intuitive concept, and the observation that Turing patterns are highly sensitive to initial conditions and variations in kinetic parameter values, the Turing mechanisms had been dismissed from the developmental community for almost two decades [3].

It was not until 1972 that Turing's idea was revived by Gierer and Meinhardt who extended and formalised Turing's ideas [4]. Despite some indications of suitable reaction-diffusion systems [5], the experimental technology available at the time was not able to convincingly identify Turing mechanisms in biological systems. It was not until another three decades later that technological advances have enabled compelling experimental evidence of Turing-like mechanisms [6]. Examples include the patterning of palatel ridges and digits, hair follicle distribution, and the patterns on the skins of animals, such as fish and zebras [7, 8, 9, 10, 11]. However, due to the complexity of the underlying systems, the exact molecular mechanisms are hard to identify most of the time, making it hard to prove that the Turing mechanism exists in nature. The sensitivity to parameters constitutes another problem as it is not clear how biological systems could have evolved to find these small parameter ranges, and how these developmental processes can be so robust to extrinsic fluctuations.

This has resulted in a large number of theoretical studies in recent years [12, 13, 14, 15, 16, 17, 18, 19]. Some recent studies have performed extensive analyses of potential network topologies [12, 14, 20]. Together these studies provide an inventory of the types of network structures that are capable of generating patterns and their robustness. The majority of these studies are preformed within a deterministic continuous framework in terms of partial differential equations (PDEs). An important question is if these results generalise to other types of modelling frameworks. This would indicate a deeper underlying principle of the Turing mechanism that is independent of the applied modelling framework. Moreover, since every model is an abstraction of a true biological system, the generalisation to different modelling frameworks would also suggest a certain robustness of the patterning processes.

Lattice gas cellular automata (LGCA) models constitute an alternative modelling framework [21]. These models discretize both space and the concentration of chemical components, and the dynamics is modelled by means of discrete diffusion and reaction steps. Turing patterns in LGCA models have first been studied in [22]. So far, only a single LGCA reaction map has been studied in this context. LGCA models have not been compared to continuous models in a broader scope with respect to Turing patterns (see also [23].

Here, we perform an exhaustive analysis of all possible two-node networks, and compare the results to the continuous modelling case. In our mathematical analysis of LGCAs we follow closely [22].

First, we analyse different network topologies with respect to Turing instabilities. Next, we analyse the emergence of patterns in simulations. It has been found in continuous models that a Turing instability is not a sufficient criterion for a pattern to manifest [20]. It remains unclear to what extent this is also true for LGCA models, or other related systems, such as Ising systems [24]. Finally, we propose a modification of the definition of Turing instabilities for LGCA models that constitutes a better predictor for patterns emerging in simulations.

This article is structured as follows. In Section 2 we introduce the mathematical description of the LGCA, its simulation procedure and the definition of Turing instabilities. Next, we describe simulation details and how to identify patterns in simulations. Subsequently, in Section 3 we present the results obtained from our analysis of two-node networks and compare these results to the continuous case. Finally, we discuss the results and conclude in Section 4.

2 Methods

In this section we present the mathematical background on lattice gas cellular automata models (LGCAs) (Section (2.1)), introduce the employed mean-field approximation (Section 2.3, define Turing instabilities (Section 2.3), and give simulation details in Section 2.4.

2.1 The LGCA model

We consider systems that consist of two interacting species modelled as discrete particles on a one-dimensional discrete lattice. The dynamics are modelled in discrete time steps that iteratively update the state of the system. Each update consists of separate reaction, shuffling, and diffusion steps, which are evaluated successively. Each spatial position consists of three compartments known as “velocity channels”, which can be either occupied or unoccupied by a single particle. Each spatial position hence can be occupied by a maximum of three particles. Accordingly, we define the state of the system at time step k and lattice position r as

$$\boldsymbol{\eta}(r,k) = (\eta_{1,1}(r,k), \eta_{1,2}(r,k), \eta_{1,3}(r,k), \eta_{2,1}(r,k), \eta_{2,2}(r,k), \eta_{2,3}(r,k)). \tag{1}$$

Each $\eta_{i,\sigma}(r,k)$ is a Boolean variable that represents the occupancy of velocity channel i of species σ, with $\eta_{i,\sigma}(r,k) = 1$ ($\eta_{i,\sigma}(r,k) = 0$) meaning the channel is occupied (unoccupied). Let further $n_\sigma(r,k)$ denote

$x = \{x_1, x_2\}$	(0,0)	(0,1)	(0,2)	(0,3)	(1,0)	(1,1)	(1,2)	(1,3)	(2,0)	(2,1)	(2,2)	(2,3)	(3,0)	(3,1)	(3,2)	(3,3)
F(x)	(0,0)	(0,0)	(0,0)	(0,0)	(3,3)	(1,1)	(0,0)	(0,0)	(3,3)	(3,3)	(2,2)	(0,0)	(3,3)	(3,3)	(3,3)	(3,3)

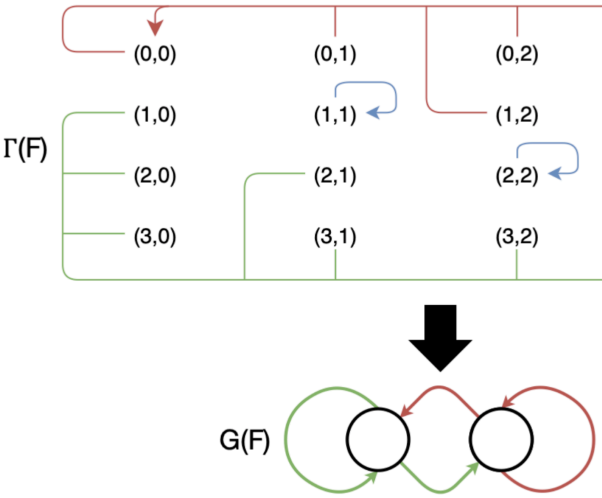


Figure 1: *The map* **F** *analysed in [23] (top panel), the map's state transition graph (middle panel) and the resulting reaction topology (bottom panel).*

the total number of particles of species σ at spatial position r and time k,

$$n_\sigma(r,k) = \sum_{i=1}^{3} \eta_{\sigma,i}(r,k). \tag{2}$$

Next, let $\mathbf{X} = \{0,1,2,3\} \times \{0,1,2,3\}$ denote the set of all possible states of $(n_1(r,k), n_2(r,k))$. We define the reaction step by a map $\mathbf{F} : \mathbf{X} \to \mathbf{X}$, where each state $\mathbf{x} \in \mathbf{X}$ is mapped onto $\mathbf{F}(\mathbf{x}) = (f_1(\mathbf{x}), f_2(\mathbf{x})) \in \mathbf{X}$. During the reaction step, each $(n_1(r,k), n_2(r,k))$ in each spatial position r is updated independently according to **F** with a probability $0 < p \leq 1$, and it remains the same with probability $(1-p)$. For $p = 1$ the reaction step becomes deterministic, and a smaller p introduces more stochasticity to the system. We refer to p as the "noise parameter".

The *state transition graph* of the map **F** is a directed graph with the set of vertices **X**, and the set of edges $\{(\mathbf{x}, \mathbf{F}(\mathbf{x})) | \mathbf{x} \in \mathbf{X}\}$ for all $\mathbf{x} \in \mathbf{X}$. See Appendix A for more details.

The *reaction topology* of the map **F**, also known as "interaction graph" in the literature [25], is a graph that summarises the interactions between species: each species is represented by a node and each interaction is represented by an edge. Each edge is assigned a positive (negative) sign if the source node of the edge acts as an activator (inhibitor) on the target node. The edges can be identified from the state transition graph as follows: for species $i, j \in \{1,2\}$ there is a positive (negative) edge from species j to i if there exists a state $\mathbf{x} = (x_1, x_2) \in \mathbf{X}$ such that $f_i(x_i, x_j + 1) - f_i(x_i, x_j)$ is positive (negative). Intuitively, this means that an increase in x_j is likely to lead to an increase (decrease) in x_i (see supplementary material S1 for more details).

Note that due to the discrete nature of the state space there exist maps $\mathbf{F} : \mathbf{X} \to \mathbf{X}$, for which there exist $\mathbf{x}, \mathbf{x}' \in \mathbf{X}$ such that $f_i(x_i, x_j+1) - f_i(x_i, x_j) > 0$ and $f_i(x'_i, x'_j+1) - f_i(x'_i, x'_j) < 0$, which means no single sign can be assigned to the edge from j to i. We omit such maps from our analysis since they cannot be assigned to a reaction topology.

Figure 1 shows the function **F**, the state transition graph, and the reaction topology for the map originally analysed in [23].

Following the reaction step, the diffusion step acts simultaneously over all lattice positions and mimics a random walk of the particles on the lattice. It is comprised of two parts: a local random shuffling, where the particles of each lattice position are randomly redistributed across the three velocity channels; this is followed by a deterministic jump step, where a particle is moved on the lattice by a predetermined amount

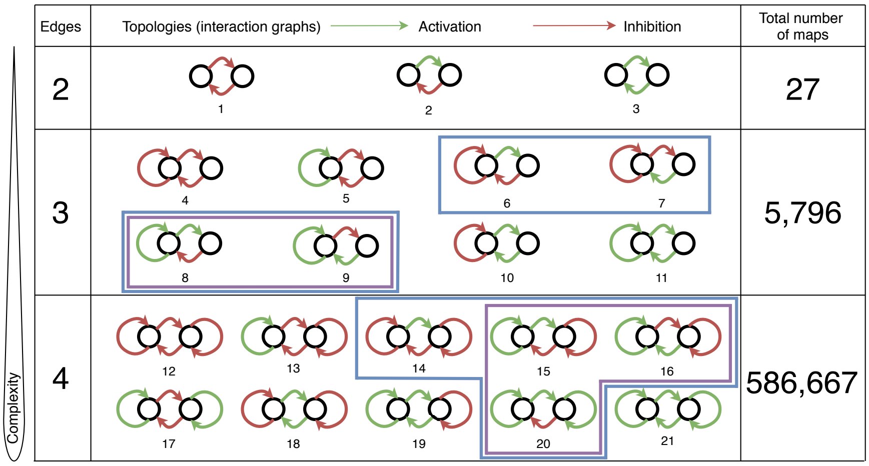


Figure 2: *The 21 two-species reaction topologies distributed across three levels of complexity, where complexity is defined by the number of edges. Highlighted in blue are the topologies found with maps that produce Turing instabilities within the LGCA framework. Highlighted in purple are the topologies found to produce Turing instabilities within the PDE framework.*

of spatial positions $d_\sigma \in \mathbb{N}_+$ in the direction associated with the velocity channel i it is occupying, $c_i \in \{-1, 0, +1\}$.

We can formalise the diffusion step by defining the difference $C_{\sigma,i}(\boldsymbol{\eta}(r,k), \boldsymbol{\eta}(r + d_\sigma c_i, k+1))$ of any given velocity channel i after one time step as

$$C_{\sigma,i}(\boldsymbol{\eta}(r,k), \boldsymbol{\eta}(r + d_\sigma c_i, k+1)) = \eta_{\sigma,i}(r + d_\sigma c_i, k+1) - \eta_{\sigma,i}(r,k). \tag{3}$$

See appendix B for more details on the diffusion step.

2.2 Mean-field approximation

So far, we have defined the dynamics of the LGCA in an algorithmic manner which is convenient to simulate stochastic simulations but not convenient for mathematical analysis, in particular in the context of pattern formation. We therefore apply a mean-field approximation to Eq. 3 which consists of neglecting correlations between channel occupations within a spatial position [23],

$$\langle \eta_{\sigma,i}(r,k)\eta_{\sigma',j}(r,k)\rangle = \langle \eta_{\sigma,i}(r,k)\rangle\langle \eta_{\sigma',j}(r,k)\rangle, \quad \text{for } i \neq j, \quad \text{and } \sigma, \sigma' \in \{1,2\}. \tag{4}$$

where $\langle\cdot\rangle$ refers to the expectation with respect to the marginal distribution of $\eta_{\sigma,i}(r,k)$.

Using this one can derive the so-called “lattice-Boltzmann equations” which describe the evolution of the expectation values of the velocity channels [22],

$$\nu_{\sigma,i}(r + d_\sigma c_i, k+1) = \nu_{\sigma,i}(r,k) + \tilde{C}_{\sigma,i}(\boldsymbol{\nu}(r,k)), \tag{5}$$

where c_i denotes channel i direction of propagation, $\mathbf{c} = \{-1, 0, 1\}$, $\boldsymbol{\nu}(r,k) = (\nu_{1,1}(r,k), \ldots, \nu_{2,3}(r,k))$, $C_{\sigma,i}(\boldsymbol{\nu}(r,k))$ is now a function of time step k only (c.f. Appendix B, Equation (17)), and we define

$$\nu_{\sigma,i}(r,k) = \langle \eta_{\sigma,i}(r,k)\rangle. \tag{6}$$

The $\nu_{\sigma,i}$ are also referred to as “single-particle distributions” and can be viewed as the probability of finding a particle of species σ in channel i on lattice position r at discrete time point k. The stochastic shuffling of the particles across the channels within the diffusion step means that the single-particle distributions for each species $\sigma \in \{1,2\}$ are indistinguishable, so we define $\nu_\sigma = \nu_{\sigma,i}, i \in \{1,2,3\}$.

2.3 Turing instabilities

We next define Turing instabilities in the context of LGCA using the lattice-Boltzmann equations in Eq.(5) by means of a linear stability analysis. A Turing instability describes the scenario where a stable steady state of a non-spatial system becomes unstable in the spatial setting including diffusion. By “stable” we mean that a small perturbation around the steady state asymptotically converges back to the steady state.

We start by defining a spatially homogeneous steady state $\bar{\nu}$ of the lattice-Boltzmann equations as the solution of

$$\nu_\sigma(r, k+1) = \nu_\sigma(r, k), \quad \tilde{C}_\sigma(\boldsymbol{\nu}(r, k)) = 0 \quad \text{for } \sigma \in \{1, 2\}. \tag{7}$$

To assess the stability of a steady state $\bar{\nu}$ we analyse the evolution of a local perturbation of a certain wavenumber q around $\bar{\nu}$ by applying a Fourier transformation to the lattice-Boltzmann equations. This decouples the individual frequency modes allowing the identification of unstable and stable modes. The evolution of the Fourier modes can be described in terms of the so-called “Boltzmann propagator” $\boldsymbol{\Gamma}(q)$ [26]. $\boldsymbol{\Gamma}(q)$ has only two non-trivial eigenvalues $\lambda_1(q)$ and $\lambda_2(q)$ which can be derived analytically (see Appendix D for the derivation and expressions).

A mode q is stable if both $|\lambda_1(q)| < 1$ and $|\lambda_2(q)| < 1$ and unstable otherwise. $q = 0$ corresponds to a homogeneous perturbation. Since diffusion does not play a role in this case (under the mean-field approximation), this corresponds to a non-spatial system. A Turing instability is hence defined as $|\lambda_1(0)|, |\lambda_2(0)| < 1$ and $|\lambda_1(q)| > 1$ or $|\lambda_2(q)| > 1$ for some $q > 0$.

If an instability exists, the wavenumber $q*$ that produces the maximum absolute value of $|\lambda(q)|$ is the fastest growing mode and the system may be expected to converge to a pattern with corresponding wavelength $\frac{L}{q}$, where L is the length of the spatial domain. For more details on the linear stability analysis see Appendix C or Chapter 13 of [23]. Figure 3a shows the absolute values of λ_1 and λ_2 as a function of wavenumber q for the map given in Figure 1.

2.4 Simulation details and power spectrum analysis

Considering all different possible combinations of positive and negative edges between the two species gives rise to 21 different fully-connected topologies shown in Figure 2 [20], i.e., topologies where both species influence each other. We only consider maps that can be assigned to one of these topologies. We thus exclude maps that have edges which cannot be assigned a positive or negative sign, as explained in Section 2.1.

We further reduce the number of maps to analyse by only considering asymptotic maps, which are defined by $f_\sigma(\mathbf{x}) \in \{0, x_\sigma, 3\}, i = 1, 2$. This corresponds to a switch-like behaviour: the total occupation number of each species gets either updated to its maximal or minimal value, or remains the same. The total number of asymptotic maps for the 21 topologies of interest is $592, 490$. Figure 2 shows the 21 possible reaction topologies grouped across three levels of complexity, which we define as the number of edges in the topology.

The formalism adopted here reduces the number of parameters considerably compared to the continuous case [20]. We set the noise parameter of the reaction step introduced in Section 2.1 to $p = 0.9$. We found empirically that $p = 1$ often leads to the system getting stuck in absorbing states that prevent Turing patterns; while smaller values of p introduce more stochasticity which tends to destroy patterns.

We further fix the diffusion constants to $(d_1, d_2) = (1, 7)$ following [23], and extensive analysis which shows that varying the diffusion parameters leads only to a negligible change in the number of maps that produce patterns. To account for the stochasticity of the system, we simulate a given map a hundred times with random initial conditions in each simulation for $T = 500$ time steps on a domain of size $L = 101$ (see Section 2.1 for the simulation details).

We then take the endpoint of each simulation, compute the power spectrum via a Fourier transform, and average the result over the different simulation runs. Finally, we fit a Lorentz distribution to the maximum of the resulting average power spectrum (see Appendix D for details). The scale parameter, γ, of the fitted Lorentz distribution corresponds to the width of the peak at half its height, and is hence a measure for the peak width. A smaller γ value indicates a sharper peak in the Fourier transform and hence a clearer pattern in the simulation result. Empirically, we choose a threshold of $\gamma < 1$ to define patterns in simulations.

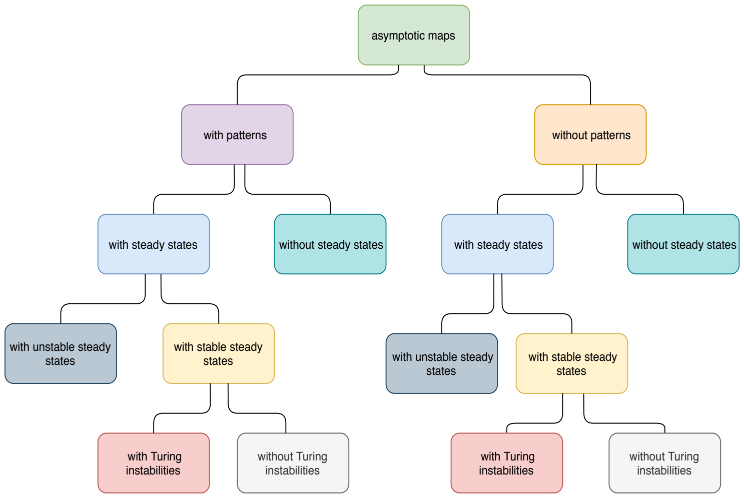
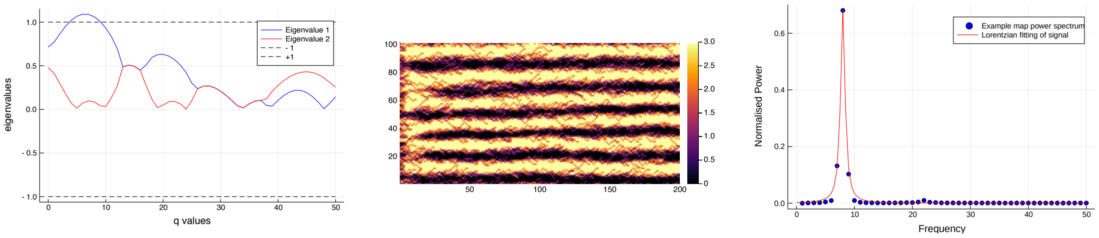

asymptotic maps
with patterns
without patterns
with steady states
without steady states
with steady states
without steady states
with unstable steady states
with stable steady states
with unstable steady states
with stable steady states
with Turing instabilities
without Turing instabilities
with Turing instabilities
without Turing instabilities

Figure 4: *Illustration of the subsets of maps. Asymptotic maps: 592,490. Maps with steady states: 338,680. Maps with stable steady states: 53,479. Turing maps: 45,591. Pattern producing maps: 20,103.*



(a)

(b)

(c)

Figure 3: *a) Dispersion relation of example map* **F**. *The plot shows that the non-spatial system is stable, that is at $q = 0$, while a clear instability exists at $q \sim 8$ for eigenvalue 1, λ_1. b) A simulation of the example map* **F** *shows a pattern emerge. c) Power spectrum of the simulation shown in figure 3b. The spectrum shows a significant signal at a frequency value of $q \sim 8$, this corresponds with figure 3a.*

3 Results

3.1 Turing instabilities

Of the $592, 490$ asymptotic maps, $51, 255$ posses a stable steady state[1]. Of these, $45, 591$ posses a Turing instability. Surprisingly, 89% of maps with a stable steady state possess also a Turing instability. Figure 4 shows the relative proportions of maps with steady states, stable steady states, Turing instabilities and those that produce patterns in simulations.

The maps with Turing instabilities are distributed across eight reaction topologies (see Figure 2, highlighted with blue). We name these topologies “Turing topologies”: all of these maps exhibit antagonistic behaviour between the two species, where the first species acts as an activator of the second species, while the second species inhibits the first. This activator-inhibitor principle is the generic mechanism used to describe Turing instabilities, first introduced by Gierer and Meinhardt [4]. Topology 8 corresponds to the classical Gierer-Meinhardt model (c.f. Figure 2), which consists of a slowly diffusing autocatalytic activator and a fast diffusing antagonist (inhibitor) species [4]. The other Turing topologies we identify are all variants of this core slow activator-fast inhibitor mechanism.

[1] Recall that Turing instabilities are only defined for maps with a stable steady state, c.f. Section 2.3

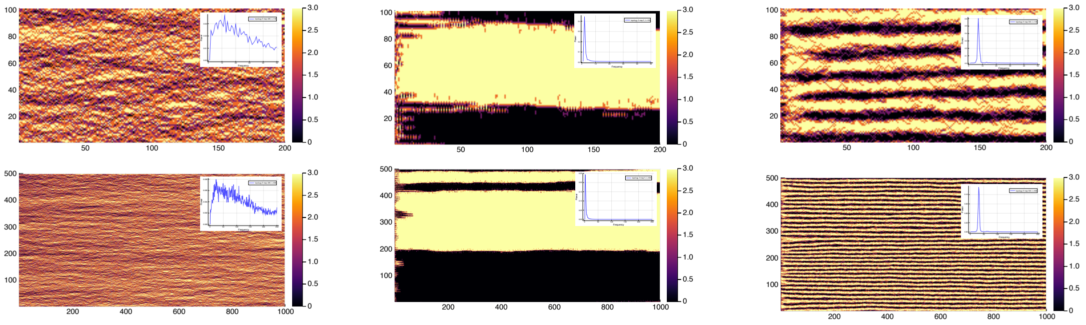
3.2 Types of simulation outcomes

Turing instabilities only describe the local instability of a steady state and do not guarantee the emergence of an actual pattern in simulations of a model. Indeed, in continuous PDE models it has been found that Turing instabilities do not always give rise to patterns in simulations [20]. Here, we find the same to be true for LGCA systems. More precisely, we observe three qualitatively different behaviours: no structure emerging, structure emerging without a characteristic wavelength (similar to "phase separation") and structure emerging with a characteristic wavelength (Turing-like patterns). We are interested in the latter: the "characteristic wavelength" of a pattern does not depend on the domain size.

Figure 5a shows an example for the first case of no structure emerging, which means that either random noise emerges (as shown in Figure 5a) or a spatially homogeneous state is reached. The corresponding power spectra do not contain significant signals indicating the absence of a pattern.

The second qualitative behaviour breaks the symmetry of the system and produces a structure without a characteristic wavelength. This behaviour is similar to the phenomenon of phase separation in PDE models [27]. The majority of these maps separates the domain into two parts, one highly expressed side and the other lowly expressed. Figure 5b shows an example of a map that breaks symmetry with no characteristic wavelength. As the domain increases so does the resulting length-scale of the structure.

The third characteristic behaviour, to which Turing patterns belong, produces a spatial structure with a characteristic wavelength that does not depend on the domain size. Figure 5c shows an example of this behaviour.



(a)

(b)

(c)

Figure 5: *(a) Example of no structure forming over varying sized spatial domains $L = 100$ (top) and $L = 500$ (bottom). (b) Example of spatial structure with no characteristic wavelength over varying sized spatial domains $L = 100$ (top) and $L = 500$ (bottom). Example of spatial structure with a characteristic wavelength over varying sized spatial domains $L = 100$ (top) and $L = 500$ (bottom). The three categories of power spectrum outcomes, a) no spatial structure, b) spatial structure without a characteristic wavelength, c) spatial structure with a characteristic wavelength.*

To automate the analysis of simulation results we proceed as explained in Section 2.4. For a simulation result to be classified as a pattern, we require the Lorentzian fitted to the normalised power spectrum averaged over n trials to have the scale $\gamma \leq \gamma^*$, i.e., the peak must not be too broad. Note that this criteria is somewhat arbitrary, as the distribution of γ values for all maps is close to continuous. We found empirically that the threshold $\gamma^* = 1$ selects only maps that produce robust clear patterning.

3.3 Turing patterns

We find that out of the $45,591$ maps with Turing instabilities, only about 33% produce a Turing pattern in simulations (see Figure 4 for a visualisation of these ratios). We thus find that a Turing instability is a rather weak predictor of a Turing pattern emerging in simulations. This is in stark contrast to the continuous case where only a tiny fraction of systems with Turing instabilities do not produce a pattern [20]. These maps are distributed across topologies 8, 9, 15, 16 and 20. Thus topologies 6, 7 and 14 are not able to produce a Turing pattern despite having a Turing instability. Instead, we find that these three topologies give rise to simulations that enter into a spatially-homogeneous limit cycle, switching between high and low homogeneous expression levels (see supplementary S3 for an example). We further found $5,285$ maps fulfilling the pattern criterion without possessing a Turing instability. These maps are distributed across the same topologies 8, 9, 15, 16 and 20 containing maps producing Turing patterns.

If we choose a larger threshold for γ, the ratio of maps with Turing instabilities that produce a pattern would increase. However, so would the number of maps without Turing instabilities producing patterns. The predictability of the Turing instability would hence not necessarily increase. Rather than using a threshold on the scale γ of the Lorentzian fit, we can alternatively view γ as an inverse measure of the quality of a pattern: the smaller γ, the sharper the peak in the power spectrum. Figure 6b shows the distribution of γ values over both maps with and without Turing instabilities. We observe that maps with Turing instabilities are more likely to result in patterns emerging in simulation data, with the majority of highest quality patterns all containing Turing instabilities.

Topology:	1	2	3	4	5	6	7	8	9	10	11
# of maps :	9	9	9	600	849	600	600	849	849	600	849
% of Turing maps:	0	0	0	0	0	0.45	0.45	0.32	0.32	0	0
% of pattern maps:	0	0	0	0	0	0	0	0.28	0.27	0	0

Table 1: *Turing instability and pattern information for topologies of complexity level 1 and 2.*

Topology:	12	13	14	15	16	17	18	19	20	21
# of maps :	40000	56600	40000	56600	56600	80089	40000	56600	80089	80089
% of Turing maps:	0	0	0.17	0.30	0.30	0	0	0	4.9	0
% of pattern maps:	0	0	0	0.15	0.15	0	0	0	0.032	0

Table 2: *Turing instability and pattern information for topologies of complexity level 3.*

3.4 Robustness

We next study the robustness of the different topologies[28], which we define as the fraction of maps of a given topology that produces Turing patterns in simulations. The robustness corresponds to the fraction in parameter space producing Turing patterns in continuously-modelled systems[29]. Figure 6a shows the ratio of Turing maps with respect to the total number of maps for each topology, and table 1 and table 2 show the total number of maps and Turing maps for each topology.

We find that for the Turing topologies a surprisingly large fraction of maps produce Turing patterns, with fractions ranging from 0.032 to 0.28. This is in stark contrast to continuously modelled systems where only small fractions of parameter space allow for Turing instabilities, with reported robustness values in kinetic parameters typically smaller than 0.01 [20]. We also find that the robustness decreases with increasing complexity of the topologies (i.e., number of edges).

3.5 No pattern despite Turing instability

As we have seen in Section 3.3, some Turing maps do not produce a pattern in simulations. This might be expected since the Turing instability only indicates a local diffusion-driven instability and does not make statements about the global behaviour. Moreover, we work with stochastic models here, which means the random fluctuations can "wash out" the wavelengths emerging from a Turing instability. Another reason is that the system can get stuck in homogeneous absorbing states. Finally, the mean-field approximation used in the stability analysis may also contribute to such discrepencies.

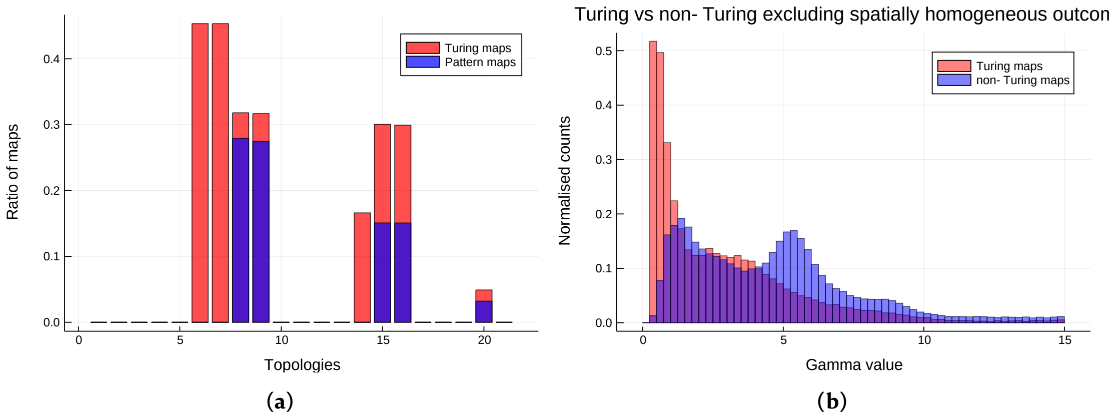


Figure 6: *a) Histogram of the ratio of maps that contain Turing instabilities and those that produce patterns. b) Comparison of the γ-value of Turing and non-Turing maps when excluding homogeneous simulation outcomes. The ratios of spatial structure producing maps for each topology and comparisons of Turing maps and non-Turing maps in terms of the quality of patterns they produce on average.*

3.6 Pattern despite no Turing instability

As discussed in Section 3.3, we find $5,285$ maps that produce a stable pattern despite not possessing a Turing instability. These are all distributed across the same topologies that possess maps with Turing instabilities that produce patterns, namely topologies 8,9,15,16 and 20. $1,011$ of the $5,285$ maps possess a stable state state and we find that these maps all have an eigenvalue whose maximal absolute value is close to one, and are hence "close" to a Turing instability (in a loose sense). One potential explanation is that fluctuations present in the model could push the system over the threshold into the instability and hence lead to a pattern of the corresponding wavelength. This has also been suggested in [22]. The remaining $4,274$ do not possess a stable steady state. Here a mechanism different to the Turing mechanism could be responsible for the pattern.

3.7 Comparison to continuous model

We next compare the results found so far using the LGCA approach with results from the literature using the continuous PDE approach.

3.7.1 Turing instabilities and emerging patterns

In Figure 2 the five two-node topologies known to produce Turing instabilities in the continuous PDE case are highlighted by purple [14, 12, 20]. Here, we find that these five topologies also produce Turing instabilities in the LGCA framework. However, we find additional Turing topologies 6, 7 and 14 (c.f. Figure 2 highlighted in blue). But these three topologies do not produce Turing patterns in simulations, despite the presence of Turing instabilities (see Section 3.2). Therefore, in terms of Turing patterns emerging in simulations, we find a 1-to-1 correspondence between Turing topologies in the LGCA and the continuous frameworks.

3.7.2 Turing instabilities as predictors of patterns

Only 33% of the maps with Turing instability produce a pattern in simulations (see 3.3; although the exact number depends on our choice of scale γ). Turing instabilities in the LGCA framework do thus not guarantee an emerging pattern in simulations. This is independent of how many stable states a map possesses. In contrast, in continuous PDE models Turing systems with a single stable steady state have empirically been found to always give rise to patterns in simulations [20]. Only Turing systems with multiple steady states have been found to sometimes converge to homogeneous steady states. Overall, Turing instabilities that do not lead to patterns in simulations seem to be substantially more frequent in the LGCA framework. We thus conclude that a Turing instability in the LGCA formalism is a weaker predictor for the existence of a pattern than in the continuous case.

3.7.3 Robustness

Even though we found the same five Turing topologies for the LGCA and continuous frameworks, we observe considerable differences in terms of their robustness. In Section 3.4 we defined the robustness of a topology as the fraction of maps that produces patterns in simulations. In the continuous modelling framework, this roughly corresponds to the fraction of kinetic parameters producing Turing patterns in a given topology (note that we do not consider robustness with respect to changes in diffusion constants or topology here as has for example been done in [20]). Importantly, we find that the topologies are substantially more robust in the LGCA framework, with robustness values ranging from 0.032 to 0.28[2], as opposed to robustness values smaller than 0.01 in the continuous-modelling case for the same two-node topologies [20].

3.8 A modified definition of Turing instabilities

Recall that we used the conventional definition of a Turing instability used in the literature [22, 23]: the absolute value of an eigenvalue of the Jacobian of the system linearised around a stable steady state becomes larger than 1 (c.f. Section 2.3). However, since a negative real part of an eigenvalue indicates an oscillatory behaviour (c.f. Equation 20), it is questionable if this is a sensible definition if one is not only interested in a stable state becoming unstable, but in the emergence of stable patterns. Indeed, topologies 6, 7 and 14 in Figure 2, for which we did not find any stable Turing patterns in simulations but only oscillatory patterns [3] despite the presence of a Turing instability, the real part of the eigenvalue in the instability is always negative. We therefore suggest that in addition to the absolute value of the eigenvalue having to be larger than 1, to include the condition that the real part of the eigenvalue is also required to be positive in the instability. Applying this modified definition of a Turing instability gets rid of oscillatory patterns, and it also leads to a 1-to-1 correspondence between Turing-instability possessing topologies in the LGCA and continuous PDE framework. Moreover, this definition of a Turing instability seems to be a better predictor of an emerging pattern: about 53% of maps with such a Turing instability produce patterns in simulations. With the original definition, this were only 33%. Positivity of the real part of the eigenvalue hence appears as a reasonable additional criterion for defining a Turing instability with the LGCA model.

4 Discussion

Recent experimental findings [7, 9, 8, 11, 10] have resulted in Turing patterns being widely accepted as an important mechanism for spatial patterning in developmental processes. These findings have raised important questions about the key features underlying the Turing mechanism and its robustness[28]. A variety of theoretical endeavours aiming to answer these questions have accordingly taken place since. However, most of these have focused on single models. More recently, some large-scale studies have systematically analysed large parts of possible design spaces, thereby providing a novel understanding on how common and robust Turing pattern mechanism are [12, 14, 20]. The majority of these theoretical studies have used differential equation models with continuous concentrations.

Since every mathematical model of a biological system is an abstraction, it is difficult to untangle those dynamics and features attributable to the true mechanics of a given biological system and those artificial dynamics arising from the modelling technique itself. Describing a biological system by different modelling frameworks can hence help to identify true underlying mechanisms; combining qualitative and quantitative

[2] We would like to point out that the exact quantitative results depend on the exact modelling details. For the LGCA case, the numbers might change if we considered all maps and not just asymptotic maps. Also, changing the threshold for the scale γ for the criterion of what we define as a pattern influences the results (c.f. Section (description of Lorentzian fitting)). Similarly, in the continuous case, the robustness values depend on the modelled parameter ranges [14, 20].

[3] Also referred to as "checkerboard patterns" in [23]

modelling frameworks[30]; or augmenting modelling by evolutionary/comparative methods can provide reassurances as to the validity of modelling studies [31].

Here, we have performed a large-scale survey of all possible two-species networks using a discrete lattice gas cellular automata (LGCA) framework. This framework is much more restrictive than the continuous approach, in that it confines the concentrations to a small number of discrete values (four in our case) and comprises discrete maps between these states rather than continuous kinetic parameters.

Using this approach, however, we found the exact same five network topologies capable of producing Turing patterns as in the continuous modelling framework [12, 14, 20]. Moreover, we found these five topologies to be even more robust in the LGCA framework than they appear to be in the continuous counterpart.

We also found that the presence of a Turing instability is neither sufficient nor necessary for pattern formation in the LGCA framework, and that it is a much weaker predictor of an emerging pattern than in the continuous counterpart. This can, to some extent, be attributed to the stochasticity of the LGCA framework which makes the definition of a pattern less straightforward, as stochasticity can wash out patterns and lead to breaking down of the employed mean-field analysis. To a certain degree this can also be attributed to the original definition of a Turing instability in the LGCA framework [22, 23], and we proposed a modified definition which leads to a stronger predictor.

The fact that our restricted model identifies the same Turing topologies identifeid before in the continuous case suggests that the exact molecular details might not be as important as previously thought and hints at a deeper underlying principle of Turing mechanisms that is independent of the modelling framework. The larger robustness we found suggests that Turing patterns might be more robust than previous continuous studies suggest.

Acknowledgements

This work has been supported through a *Life?* programme grant from the *Volkswagen Stiftung* to HS and MHPS; TL gratefully acknowledges a PhD scholarship from the EPSRC; and DS and MPHS thank the BBSRC for funding (BB/R000816/1). HS acknowledges financial support through the DFG Forschungszentrum MATHEON initiative. MPHS is funded through the University of Melbourne *Driving Research Momentum* program.

A Reaction step

Following from section 2.1 we derive the general expression for the channel after the reaction step. Recall from section 2.1 that the total number of particles of a given species σ at a given position r and time k is

$$n_\sigma(\boldsymbol{\eta}_\sigma)(r,k) = \sum_{i=1}^{3} \eta_{\sigma,i}(r,k). \tag{8}$$

After the reaction step the newly updated total number of particles of species σ at a given r and time k is

$$n_\sigma^R(\boldsymbol{\eta}(r,k)) = \sum_{i=0}^{3}\sum_{j=0}^{3} \Psi_i(\boldsymbol{\eta}_1(r,k))F_\sigma(i,j)\Psi_j(\boldsymbol{\eta}_2(r,k)), \tag{9}$$

where $\Psi(\boldsymbol{\eta}_\sigma(r,k)) = (\psi^0(\boldsymbol{\eta}_\sigma(r,k)), \psi^1(\boldsymbol{\eta}_\sigma(r,k)), \psi^2(\boldsymbol{\eta}_\sigma(r,k)), \psi^3(\boldsymbol{\eta}_\sigma(r,k)))$ with $\psi^i(\boldsymbol{\eta}_\sigma(r,k)) = 1$ when i particles of species σ exist at the lattice position r and 0 otherwise. See supplementary S4 for a full expression of $\Psi(\boldsymbol{\eta}_\sigma(r,k))$. $\mathbf{F}_\sigma(i,j)$ denotes the updated value of species σ from the state $\mathbf{x} = (i,j)$ as defined in section 2.1. The superscript ^{R} indicates the variable after the reaction step. We add stochasticity to the interaction step by introducing sequences of spatial and time independent identically distributed Bernoulli random variables $\epsilon \in \{0,1\}, r \in \mathcal{L}, k \in \mathbb{N}$. These variables determine whether the reaction takes place or not. We further define $p = P(\epsilon(r,k) = 1)$, where p is the probability of the reaction taking place. We refer to p as the "noise parameter". With these definitions the post-reaction total number of particles of species σ at a given r and time k reads

$$n_\sigma^R(\boldsymbol{\eta}(r,k)) = (1-\epsilon)n_\sigma(r,k) + \epsilon \sum_{i=0}^{l}\sum_{j=0}^{l} \Psi^i(\boldsymbol{\eta}_1(r,k))\mathbf{F}_\sigma(i,j)\Psi^j(\boldsymbol{\eta}_2(r,k)), \tag{10}$$

Once the total number of particles n_σ^R are updated they are redistributed back into the individual channels $\eta_{\sigma,i}^R$ for all i, such that

$$\eta_{\sigma,i}^R(r,k) = \begin{cases} 0, & \text{if } n_\sigma^R(r,k) < i. \\ 1, & \text{otherwise.} \end{cases} \tag{11}$$

B Shuffling step, diffusion step and difference equation

In this section we provide a formal definition of the shuffling and diffusion steps described in Section 2.1, and give the full expression for the difference equation in (3). We start by expressing the random shuffling step in terms of permutation matrices. The set of permutation matrices for a system with three velocity channels is

$$\mathcal{A}_3 = \{ \begin{bmatrix}1&0&0\\0&1&0\\0&0&1\end{bmatrix}, \begin{bmatrix}1&0&0\\0&0&1\\0&1&0\end{bmatrix}, \begin{bmatrix}0&1&0\\1&0&0\\0&0&1\end{bmatrix}, \begin{bmatrix}0&0&1\\0&1&0\\1&0&0\end{bmatrix}, \begin{bmatrix}0&1&0\\0&0&1\\1&0&0\end{bmatrix}, \begin{bmatrix}0&0&1\\1&0&0\\0&1&0\end{bmatrix} \}. \tag{12}$$

Using these we can write the updated channels in terms of the set of local channels at spatial position r as

$$\eta_{\sigma,i}^{\mathcal{R}}(r,k) = \sum_{j=1}^{6} \zeta_j(r,k) \sum_{l=1}^{3} \eta_{\sigma,l}^R(r,k) a_{li}^j, \tag{13}$$

where the superscript $\mathcal{R}$ indicates the variable after the shuffling step. a_{li}^j denotes the element of the j^{th} permutation matrix at row l and column i, and $\zeta_j \in \{0,1\}$, $j \in 1, \dots 6$ are Bernoulli type random variables, such that $\zeta_j = 1$ for one $j \in \{1,\dots,6\}$ and zero otherwise. After the shuffling step we apply the jump diffusion step as

$$\eta_{\sigma,i}(r + d_\sigma c_i, k+1) = \eta_{\sigma,i}^{\mathcal{R}}(r,k). \tag{14}$$

Combining Equations (3), (13) and (14) we obtain an expression for the change in expression level over a single time step, $C_{\sigma,i}(\boldsymbol{\eta}(r,k), \boldsymbol{\eta}(r+d_\sigma, k+1))$, in terms of the variables of the previous time step:

$$\begin{aligned} C_{\sigma,i}(\boldsymbol{\eta}(r,k), \boldsymbol{\eta}(r+d_\sigma,k+1)) &= \eta_{\sigma,i}(r+d_\sigma,k+1) - \eta_{\sigma,i}(r,k) && (15)\\ &= \eta_{\sigma,i}^{\mathcal{R}}(r,k) - \eta_{\sigma,i}(r,k) && (16)\\ &= \sum_{j=1}^{6} \zeta_j(r,k) \sum_{l=1}^{3} \eta_{\sigma,l}^R(r,k) a_{li}^j - \eta_{\sigma,i}(r,k). && (17) \end{aligned}$$

and we can write $C_{\sigma,i}(\boldsymbol{\eta}(r,k), \boldsymbol{\eta}(r+d_\sigma, k+1)) = C_{\sigma,i}(\boldsymbol{\eta}(r,k))$. Equation (17) describes how each individual channel evolves over time.

C Linear stability analysis in the LGCA model

Here, we derive the equations needed for the stability analysis used to define Turing instabilities in the mean-field approximation in Section 2.3. As mentioned in Section 2.3, we use a small perturbation to determine the stability of a given steady state: let $\delta\nu_i(r,k) = \nu_i(r,k) - \bar{\nu} \in \mathbb{R}^6$ be a small perturbation around the steady-state solution $\bar{\nu}$ of Equation (7), where i denotes the channel. Using this in (5) and linearizing around $\bar{\nu}$ we get the linear lattice-Boltzmann equation

$$\delta\nu_i(r + d^{(i)}c^{(i)}, k+1) = \delta\nu_i(r,k) + \sum_{j=1}^{6} \Omega_{ij}\delta\nu_j(r,k), \tag{18}$$

where the diffusion coefficient $d^{(i)}$ is the ith element of $(d_{\sigma=1}, d_{\sigma=1}, d_{\sigma=1}, d_{\sigma=2}, d_{\sigma=2}, d_{\sigma=2})$ and the direction $c^{(i)}$ is the ith element of $(1, 0, -1, 1, 0, -1)$. The Jacobian $\in \mathbb{R}^{6\times 6}$ is defined as

$$\Omega_{ij} = \frac{\partial \tilde{C}_i(\delta\nu(r,k))}{\partial \delta\nu_j(r,k)}\Bigg|_{\nu} = \bar{\nu}. \tag{19}$$

Consider a harmonic wave perturbation of the form

$$\delta\nu_i(r,k) \propto \lambda^k \cos(\frac{\pi}{L}qr), \tag{20}$$

where $q = 0$ corresponds to a spatially homogeneous perturbation. Next, we consider a general perturbation $\mathbf{F}(q,k) = (F_1(q,k), \ldots, F_6(q,k))$ and express each of its components as a sum of sinusoidal terms as

$$F_i(q,k) = \sum_{r\in L} \delta\nu_i(r,k)e^{\frac{2\pi i}{L}qr}. \tag{21}$$

Applying the discrete Fourier transformation to the linear lattice-Boltzmann equations (18) gives

$$F_i(q, k+1) = e^{-\frac{2\pi i}{L}qd^{(i)}c^{(i)}}\left(F_i(q,k) + \sum_{j=1}^{6}\Omega_{ij}F_j(q,k)\right), \tag{22}$$

which we write in vectorised from for as

$$\mathbf{F}(q,k+1) = \mathbf{\Gamma}(k)\mathbf{F}(q,k), \forall q, \tag{23}$$

where $\mathbf{\Gamma}(k)$ is the Boltzmann propagator defined by

$$\Gamma_{ij}(q) = e^{\frac{-2\pi i}{L}qd^{(i)}c^{(i)}}\left\{\delta_{ij} + \Omega_{ij}\right\}. \tag{24}$$

In matrix notation this can be written as

$$\mathbf{\Gamma}(q) = T(I + \mathbf{\Omega}). \tag{25}$$

where $\mathbf{T} \in \mathbb{R}^{6\times 6}$ which is known as the "Transport matrix" and defined as a diagonal matrix with elements $\mathbf{T}_{jj} = e^{\frac{2\pi i}{L}qd_jc_j}$, and $\mathbf{I} \in \mathbb{R}^{6\times 6}$ is the identity matrix. $(\mathbf{I} +\ )$ is a block matrix and reads

$$\{I + \mathbf{\Omega}\} = \begin{bmatrix} \omega_1 & \omega_1 & \omega_1 & \omega_2 & \omega_2 & \omega_2 \\ \omega_1 & \omega_1 & \omega_1 & \omega_2 & \omega_2 & \omega_2 \\ \omega_1 & \omega_1 & \omega_1 & \omega_2 & \omega_2 & \omega_2 \\ \omega_3 & \omega_3 & \omega_3 & \omega_4 & \omega_4 & \omega_4 \\ \omega_3 & \omega_3 & \omega_3 & \omega_4 & \omega_4 & \omega_4 \\ \omega_3 & \omega_3 & \omega_3 & \omega_4 & \omega_4 & \omega_4 \end{bmatrix}, \quad \omega_1, \omega_2, \omega_3, \omega_4 \in \mathbb{R}. \tag{26}$$

See supplemental SXX for expressions for $\omega_1, \omega_2, \omega_3$ and ω_4. We define $\mathbf{\Lambda}_{\Gamma(q)} = (\lambda_1(q), \lambda_2(q), \lambda_3(q), \lambda_4(q), \lambda_5(q), \lambda_6(q))$, where $\lambda_i(q)$ is the ith eigenvalue of $\mathbf{\Gamma}(q)$. The λ_i determines the stability of a given steady state and are obtained as solutions of

$$|\mathbf{\Gamma}(q) - \lambda I| = 0, \tag{27}$$

where $|\cdot|$ denotes the determinant. Due to the block structure of $\mathbf{\Gamma}(q)$ (c.f. equation 26) only two of the eigenvalues are non-zero: $\mathbf{\Lambda}_{\Gamma(q)} = (\lambda_1(q), \lambda_2(q), 0, 0, 0, 0)$, with

$$\lambda_{1,2}(q) = \frac{1}{2}\left(\omega_1 u_1(q) + \omega_4 u_2(q) \pm \sqrt{4(\omega_2\omega_3 - \omega_1\omega_4)u_1(q)u_2(q) + (\omega_1 u_1(q) + \omega_4 u_2(q))^2}\right), \tag{28}$$

where $u_\sigma(q) = 1 + e^{-\frac{2\pi i}{L}qd_\sigma} + e^{\frac{2\pi i}{L}qd_\sigma}$. When $q = 0$, λ_1 and λ_2 define the stability of the non-spatial system.

D Power spectrum analysis

In this section we provide details on how we use the power spectrum in Section 2.4 to automatically identify patterns in simulations as outlined in Section 2.4. Since either both or none of the two species show patterns in simulations, we only use the number of particles $n_1(r, T)$ of the first species defined in Equation (2) at end time point T to identify patterns. The normalised power spectrum of a simulation run as described in Section 2.4 is given by

$$S(q) = \left| \sum_{r=0}^{L-1} n_1(r, T) e^{\frac{-2\pi i q r}{L}} \right|^2 . \tag{29}$$

To average out fluctuations we perform K simulation trials. Let $S(q)^{(i)}, i = 1, \dots, K$ be the corresponding power spectra computed as in Equation (29). We accordingly define $\bar{S}(q)$ as the average over the K trials:

$$\bar{S}(q) = \frac{1}{K} \sum_{j=1}^{K} S(q)^{(j)} . \tag{30}$$

If $\bar{S}(q)$ possesses a clear peak at a certain wavenumber q this indicates a spatial pattern in the simulation results with wavelength L/q. To identify such a peak we fit a Lorentz distribution to $\bar{S}(q)$ to quantify the quality of a pattern. The Lorentz distribution is defined as

$$f(x, x_0, \gamma) = \frac{1}{\pi\gamma[1 + (\frac{x-x_0}{\gamma})^2]} \tag{31}$$

[32]. We determine the median x_0 by finding the frequency in which the cumulative distribution function is equal to 0.5. We then use the method of least squares to determine the value of the scale parameter $\gamma \in [0, 1]$. The smaller the scale parameter the sharper the peak and hence the clearer the pattern in the spatial domain. We thus use a threshold on γ to judge if a simulation result contains a pattern or not, as explained in Section 2.4.

References

[1] A. M. Turing, "The chemical basis of morphogenesis," *Bulletin of mathematical biology*, vol. 52, no. 1-2, pp. 153–197, 1990.

[2] S. Kondo and T. Miura, "Reaction-diffusion model as a framework for understanding biological pattern formation," *science*, vol. 329, no. 5999, pp. 1616–1620, 2010.

[3] L. Marcon and J. Sharpe, "Turing patterns in development: what about the horse part?," *Curr Opin Genet Dev*, vol. 22, pp. 578–84, Dec 2012.

[4] A. Gierer and H. Meinhardt, "A theory of biological pattern formation," *Kybernetik*, vol. 12, no. 1, pp. 30–39, 1972.

[5] H. Meinhardt, "Models of biological pattern formation," *New York*, p. 118, 1982.

[6] C. Economou, T. Wannathong, J. Szaub, and S. Purton, "A simple, low-cost method for chloroplast transformation of the green alga chlamydomonas reinhardtii," in *Chloroplast Biotechnology*, pp. 401–411, Springer, 2014.

[7] S. Raspopovic, M. Capogrosso, F. M. Petrini, M. Bonizzato, J. Rigosa, G. Di Pino, J. Carpaneto, M. Controzzi, T. Boretius, E. Fernandez, *et al.*, "Restoring natural sensory feedback in real-time bidirectional hand prostheses," *Science translational medicine*, vol. 6, no. 222, pp. 222ra19–222ra19, 2014.

[8] S. Sick, S. Reinker, J. Timmer, and T. Schlake, "Wnt and dkk determine hair follicle spacing through a reaction-diffusion mechanism," *Science*, vol. 314, no. 5804, pp. 1447–1450, 2006.

[9] H.-S. Jung, P. H. Francis-West, R. B. Widelitz, T.-X. Jiang, S. Ting-Berreth, C. Tickle, L. Wolpert, and C.-M. Chuong, "Local inhibitory action of bmps and their relationships with activators in feather formation: implications for periodic patterning," *Developmental biology*, vol. 196, no. 1, pp. 11–23, 1998.

[10] A. Nakamasu, G. Takahashi, A. Kanbe, and S. Kondo, "Interactions between zebrafish pigment cells responsible for the generation of turing patterns," *Proceedings of the National Academy of Sciences*, vol. 106, no. 21, pp. 8429–8434, 2009.

[11] A. D. Economou, A. Ohazama, T. Porntaveetus, P. T. Sharpe, S. Kondo, M. A. Basson, A. Gritli-Linde, M. T. Cobourne, and J. B. Green, "Periodic stripe formation by a turing mechanism operating at growth zones in the mammalian palate," *Nature genetics*, vol. 44, no. 3, pp. 348–351, 2012.

[12] L. Marcon, X. Diego, J. Sharpe, and P. Müller, "High-throughput mathematical analysis identifies turing networks for patterning with equally diffusing signals," *Elife*, vol. 5, p. e14022, 2016.

[13] X. Diego, L. Marcon, P. Müller, and J. Sharpe, "Key features of turing systems are determined purely by network topology," *Physical Review X*, vol. 8, no. 2, p. 021071, 2018.

[14] M. M. Zheng, B. Shao, and Q. Ouyang, "Identifying network topologies that can generate turing pattern," *Journal of theoretical biology*, vol. 408, pp. 88–96, 2016.

[15] A. D. Economou, N. A. Monk, and J. B. Green, "Perturbation analysis of a multi-morphogen turing reaction-diffusion stripe patterning system reveals key regulatory interactions," *bioRxiv*, pp. 2019–12, 2020.

[16] M. Merle, L. Messio, and J. Mozziconacci, "Turing-like patterns in an asymmetric dynamic ising model," *Physical Review E*, vol. 100, no. 4, p. 042111, 2019.

[17] S. Smith and N. Dalchau, "Model reduction enables turing instability analysis of large reaction–diffusion models," *Journal of The Royal Society Interface*, vol. 15, no. 140, p. 20170805, 2018.

[18] Q. Zheng and J. Shen, "Turing instability induced by random network in fitzhugh-nagumo model," *Applied Mathematics and Computation*, vol. 381, p. 125304, 2020.

[19] Y. Chen and J. Buceta, "A non-linear analysis of turing pattern formation," *PloS one*, vol. 14, no. 8, p. e0220994, 2019.

[20] N. S. Scholes, D. Schnoerr, M. Isalan, and M. P. Stumpf, "A comprehensive network atlas reveals that turing patterns are common but not robust," *Cell systems*, vol. 9, no. 3, pp. 243–257, 2019.

[21] D. A. Wolf-Gladrow, *Lattice-gas cellular automata and lattice Boltzmann models: an introduction*. Springer, 2004.

[22] S. Dormann, A. Deutsch, and A. T. Lawniczak, "Fourier analysis of turing-like pattern formation in cellular automaton models," *Future Generation Computer Systems*, vol. 17, no. 7, pp. 901–909, 2001.

[23] A. Deutsch, S. Dormann, *et al.*, *Cellular automaton modeling of biological pattern formation*. Springer, 2005.

[24] M. Merle, L. Messio, and J. Mozziconacci, "Turing-like patterns in an asymmetric dynamic Ising model.," *Physical Review E*, vol. 100, p. 042111, Oct. 2019.

[25] A. Richard and J.-P. Comet, "Necessary conditions for multistationarity in discrete dynamical systems," *Discrete Applied Mathematics*, vol. 155, no. 18, pp. 2403–2413, 2007.

[26] U. Frisch, D. d'Humieres, B. Hasslacher, P. Lallemand, Y. Pomeau, J.-P. Rivet, *et al.*, "Lattice gas hydrodynamics in two and three dimensions," *Complex systems*, vol. 1, no. 4, pp. 649–707, 1987.

[27] A. A. Hyman, C. A. Weber, and F. Jülicher, "Liquid-liquid phase separation in biology," *Annual review of cell and developmental biology*, vol. 30, pp. 39–58, 2014.

[28] P. K. Maini, T. E. Woolley, R. E. Baker, E. A. Gaffney, and S. S. Lee, "Turing's model for biological pattern formation and the robustness problem," *Interface Focus*, vol. 2, pp. 487–496, June 2012.

[29] P. D. W. Kirk, D. M. Y. Rolando, A. L. Maclean, and M. P. H. Stumpf, "Conditional random matrix ensembles and the stability of dynamical systems," *New Journal of Physics*, vol. 17, p. 083025, Aug. 2015.

[30] T. Toni, G. Jovanovic, M. Huvet, M. Buck, and M. P. H. Stumpf, "From qualitative data to quantitative models: analysis of the phage shock protein stress response in Escherichia coli.," *Bmc Systems Biology*, vol. 5, p. 69, May 2011.

[31] M. Huvet, T. Toni, X. Sheng, T. W. Thorne, G. Jovanovic, G. Jovanovic, C. Engl, M. Buck, J. W. Pinney, and M. P. H. Stumpf, "The evolution of the phage shock protein response system: interplay between protein function, genomic organization, and system function.," *Molecular Biology and Evolution*, vol. 28, pp. 1141–1155, Mar. 2011.

[32] G. C. King, *Vibrations and waves*. John Wiley & Sons, 2013.