KTH ROYAL INSTITUTE
OF TECHNOLOGY

Degree Project in Technology

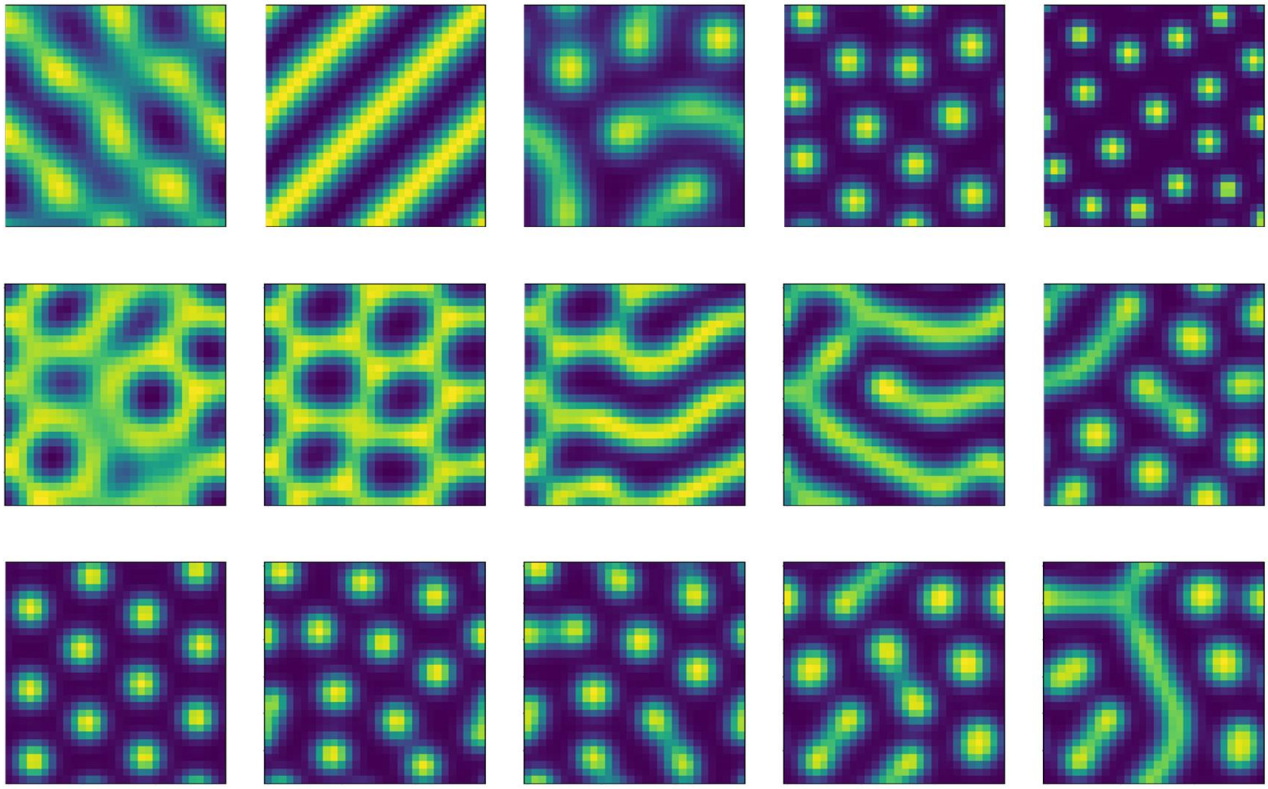
First cycle, 15 credits

Turing’s model for pattern formation

A study of pattern characteristics and minimal energy control

OSKAR FALGÉN NIKULA, OSKAR FORSSTRÖM

SUPERVISORS: YIBEI LI, YUEXIN CAO



Stockholm, Sweden 2022

Abstract (EN)

In an attempt to describe how patterns emerge in biological systems, Alan Turing proposed a mathematical model encapsulating the properties of such processes. It details a partial differential equation governing the dynamics of two or more substances, called morphogens, reacting and diffusing in a specific manner, in turn generating what has now come to be denoted as Turing patterns. In recent years, evidence has accumulated to support Turing's claim and it has been proposed that it is responsible for the dynamical characteristics of phenomena such as skin pigmentation and branching of lungs in vertebrates. The aim of this paper is to study how the choice of model parameters and reaction kinetics influence the nature of patterns generated, as well as explore how boundary control can be employed to generate pre-defined patterns and the efficiency of this procedure. To simulate the patterns, the differential equation is solved in Python by means of a spectral method using discretized space and time domains. The model parameters were then studied to try to gain insight in their effects on the patterns yielded. The boundary control was implemented in MATLAB using a difference method. The metric used for efficiency was taken to be the energy expenditure of the boundary cells. The complex dynamics of the studied systems make it difficult to draw valuable conclusions on the influence of the parameters, but the results support the expected characteristics of the models used. The efficiency of the pattern generation is deemed to be closely related to the amount of boundary control utilized.

KEYWORDS: Reaction-diffusion, Turing pattern, morphogen, Grier-Meinhardt, Schnakenberg, numerical analysis, minimum energy control.

Abstract (SV)

I ett försök att beskriva hur mönster bildas i biologiska system föreslog Alan Turing en matematisk modell som inbegriper egenskaperna hos sådana processer. Modellen utgörs av en partiell differentialekvation som karaktäriserar dynamiken av två eller fler ämnen, s.k. morfogener, som reagerar och diffunderar enligt vissa krav på ett sådant sätt att s.k. Turing-mönster bildas. Med åren har bevis framlagts som understöder Turings ansats och det har föreslagits att den kan förklara de dynamiska processerna hos fenomen såsom hudpigmentering samt förgrening av lungor i ryggradsdjur. Detta arbete ämnar utforska hur valet av modellparametrar och -reaktionskinetik påverkar beskaffenheten hos de mönster som genereras, samt att studera hur randvärdeskontroll kan implementeras för att generera förutbestämda mönster och utforska effektiviteten av denna process. För att simulera mönster löses differentialekvationen i Python genom implementering av en spektralmetod i diskretiserat rums- och tidsdomäner. Modellparametrarna undersöktes för att få förståelse kring deras inverkan på de mönster som erhålls. Randvärdeskontrollen implementerades med hjälp av en finit differensmetod i MATLAB där måttstocken för effektivitet valdes som energiåtgången hos randcellerna. Systemets komplexa dynamik gjorde det svårare att dra värdefulla slutsatser om modellparametrarnas inverkan på mönsterbildningen, men resultaten stödjer de förväntade egenskaperna hos de modeller som användes. Det fastslogs att effektiviteten med vilken mönster genererades var starkt sammankopplat med randvärdeskontrollens omfattning.

NYCKELORD: Reaktion-diffusion, Turing-mönster, morfogen, Grier-Meinhardt, Schnakenberg, numerisk analys, minimalenergikontroll

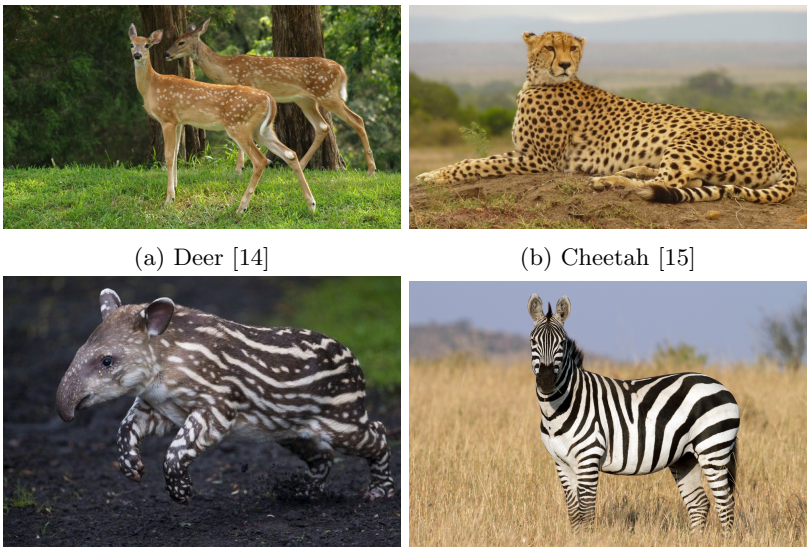
Contents

1 Introduction

In 1952, in his paper "The chemical basis of morphogenesis", Alan Turing proposed a theory for the biochemical mechanisms responsible for pattern formation in organisms. The model, a so-called reaction-diffusion model, describes the reaction between and diffusion of two substances, called morphogens. Turing hypothesized that initially, the system is spatially homogeneous and is resistant to external perturbations. Interestingly, he proposed that given a minuscule perturbation from this state, diffusion would drive the system unstable, moving it from its homogeneous steady state, leading to a non-homogeneous, spatiotemporally stable pattern through the kinetics of the two substances. In this paper, the model will be explored and used to simulate patterns akin to those that emerge in nature, such as the pigmentation in marine animals.

The purpose of this model is to try to describe biological patterning processes and although it is a grave simplification of the real world phenomena it is trying to describe, observations have been made that support Turing's theory. Skin patterning in fish have been replicated by means of simulation and the dynamic properties that govern these patterns have been attributed to the interaction between pigment cells, as described by Turing's model [6]. Other processes also show promising dynamical characteristics, such as limb formation [11] and the branching of lungs in vertebrates [6], but these have yet to be confirmed to be governed by the Turing model.

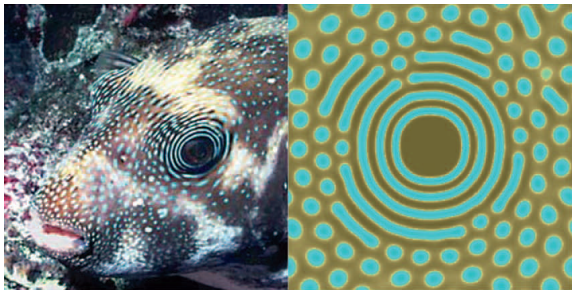
Grier and Meinhardt found that the only condition required on a system to produce Turing patterns is a short-range positive feedback, long-range positive feedback [3] [2]. This will be explained in further detail. This broadens the scope of applications of the Turing pattern model. It is actually not even required for the mode of transmission to be diffusion, there are other viable candidates as well. Signaling phenomena such as chemotactic cell migration (movement of cells due to chemical stimulus) [7], mechanochemical activity (activity brought about by the interplay of mechanical and chemical energy) [4] and neuronal interactions [13] are all possible nominees. All of these effects can give rise to periodic patterns given that the short-range positive feedback, long-range positive feedback condition is satisfied. Revisiting the case of skin patterning in fish, although experimental evidence points toward diffusion not being the mode of transmission responsible for the patterning process, the reaction-diffusion Turing model can predict how the pattern moves during fish growth [9].



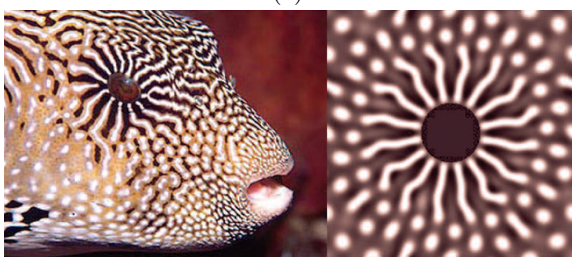
(a) Deer [14] (b) Cheetah [15]

(c) Tapir [8] (d) Zebra [12]

Figure 1: Examples of furred animals whose patterns may stem from processes similar to those posed by the Turing model.



(a)



(b)

Figure 2: Examples of fish pigmentation patterns that have been recreated by means of the Turing model. [6]

There are many aspects of the Turing model and great diversity in its applications, which breeds a plethora of possible points of study. To limit the scope of this thesis, the questions to be answered are

1. How does the choice of reaction model and its parameters affect the appearance of the generated patterns?

2. How does the energy expenditure relate to the amount of control over the boundary?

Analysis of the Turing PDE was accomplished by means of numerical PDE solvers. To answer the first question, an analysis was carried out with two different kinds of models of reaction. The reaction models' associated parameters were varied, whilst closely observing the resulting visual differences in the generated Turing patterns, leading to conclusions about the parameters' influence on the appearance of Turing patterns. The second question was answered by studying boundary value control of the Turing model, formulating and solving a minimization problem of the boundary value control, with the objective of minimizing a defined cost-function whose value could be interpreted as the amount of expended energy. From the ensuing results, it was concluded that the energy expenditure of boundary value control can be drastically decreased by increasing the amount of control over the boundary.

This report is structured as follows. In section 2, some fundamental theory is presented about both the Turing model and involved numerical methods. Also specifics about the implementation is described. Subsequently, in section 3 the results obtained in the analysis is presented and discussed in relation to the questions posed in this section. Finally, we conclude and summarize the report in section 4.

2 Theory and Method

2.1 Turing's Model

Turing proposed a model for the emergence of patterns through reaction and diffusion of two substances, called morphogens. To model the mechanisms of such a system, Turing proposed the following equation [5]

$$\frac{\partial \mathbf{u}}{\partial t} = \mathbf{D}\nabla^2\mathbf{u} + \mathbf{f}(\mathbf{u}), \quad \mathbf{f}(\mathbf{u}) = \begin{bmatrix} f(u,v) \\ g(u,v) \end{bmatrix}, \quad \mathbf{u} = \begin{bmatrix} u \\ v \end{bmatrix}, \quad \mathbf{D} = \begin{bmatrix} D_u & 0 \\ 0 & D_v \end{bmatrix} \tag{1}$$

subject to boundary and initial conditions

where each element in the vector quantity, $\mathbf{u}$, denotes the concentrations of each morphogen. $\mathbf{D}$ is the diffusion matrix and will hereby be defined to be diagonal. Its elements are the diffusion coefficients of u and v, denoted D_u and D_v, respectively. The equation can be interpreted as the diffusion equation with an added $\mathbf{f}(\mathbf{u})$ term describing the reaction of the two substances. This can take on many different forms, since it models the specific so-called reaction kinetics of the system. That is, the rates of reactions and reaction topology. In this paper, two such functions will be examined. The first being the Grier-Meinhardt kinetics [5].

$$\begin{cases} f(u,v) = c_1 - c_2 u + c_3 \frac{u^2}{(1+ku^2)v} \\ g(u,v) = c_4 u^2 - c_5 v \end{cases} \tag{2}$$

u denotes what is called an activator, promoting the synthesis of v and itself. v is an inhibitor, inhibiting the production of u. $\frac{c_3}{(1+ku^2)v}$ describes the rate of self-activation of u and also encapsulates the inhibiting effect of v on u. c_4u^2 describes the activation of v by u at rate c_4. c_2 and c_5 correspond to the degradation of the respective morphogen and c_1 is a constant describing the rate of synthesis of u stemming from the reaction of a substrate or a constant feeding of u to the system in general. The following equations shows a chemical reaction interpretation of the Grier-Meinhardt kinetics.

$$u + u \xrightarrow[\text{inhibition}]{c_3 3/(1+ku^2)v} 3u, \quad u + u \xrightarrow[\text{activation}]{c_4} 2u + v.$$

The second function is the Schnakenberg kinetics

$$\begin{cases} f(u,v) = c_1 - c_{-1}u + c_3 u^2 v \\ g(u,v) = c_2 - c_3 u^2 v \end{cases} . \tag{3}$$

This is a so-called substrate-activator model. It was introduced by Schnakenberg as the simplest possible model capable of forming Turing patterns [1]. u and v reside in a substrate of two substances A and B. A can react to produce u and u can degrade to form A again. These rates correspond to parameter c_1 and c_{-1}, respectively. B reacts to form v at rate c_2. Also, two molecules of u can react with one molecule of v to form three molecules of u at the reaction rate of c_3, which describes the last terms in both expressions of equation (3) [1]. Importantly, in the Grier-Meinhardt model, the activator exhibits short-range action, while the inhibitor acts on long range. This dynamic also holds true for u and v in the Schnakenberg model. The practial effect of this is that $D_u << D_v$. The following equations is a chemical reaction intepretation of the Schnakenberg reaction function.

$$u \underset{c_1}{\overset{c_{-1}}{\rightleftharpoons}} A, \quad B \xrightarrow{c_2} v, \quad 2u + v \xrightarrow{c_3} 3u.$$

The parameters in each of these models define a parameter space Φ, but not all choices of parameters yield Turing patterns. A Turing pattern is a non-homogenenous pattern that is insensitive

to initial conditions and temporally stable once it has settled. A common procedure for generating Turing patterns involves three criteria. These are not necessary conditions for Turing patterns to form, but serve as a good practice. Firstly, the initial morphogen concentrations are chosen to be a homogeneous steady state. This is a stable equilibrium for equation (1), disregarding the diffusion term. Phrased differently, it is a constant value for $\mathbf{u}$, that makes equation (1) zero-valued. In other words, the time derivative is zero and the concentrations will be unchanged with time. Since $\nabla^2 C = 0$ for constant C, the only condition that needs to be satisfied is

$$\mathbf{f}(\mathbf{u}^*) = 0 \tag{4}$$

$\mathbf{u}^*$ is called the homogeneous steady state. Secondly, one should find parameters such that the spatial system is locally unstable. Turing then found that if the system is offset slightly from $\mathbf{u}^*$, diffusion, that is normally a homogenizing phenomenon, can actually drive the system into instability, leading to pattern formation. This is called diffusion driven instability (DDI) and constitutes the second criterion as follows

$$\begin{cases} f_u + g_v < 0 \\ f_u g_v - f_v g_u > 0 \\ D_u g_v + D_v f_u > 2\sqrt{D_u D_v}\sqrt{f_u g_v - f_v g_u} > 0 \end{cases}, \tag{5}$$

where f_u, g_u, f_v and g_v denote partial derivatives with respect to u and v, respectively. These inequalities are evaluated at $\mathbf{u}^*$, thus making them conditions on the parameters. These two criteria define a subset $\Phi_0 \subseteq \Phi$ of the total parameter space that shuold permit the production of spatial Turing patterns, but this should be verified by simulation. To reiterate, there must be a short-range activation/long-range inhibition relationship between the two morphogens. As noted above, this means the diffusion coefficient for the activator must be smaller than that of the inhibitor. Below is a flowchart of the described pattern generation procedure. [11]

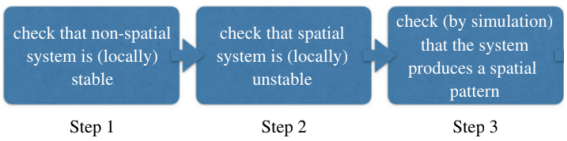


Figure 3: Flow chart of Turing pattern generation procedure.

2.2 PDE Solvers

Equation (1) is a non-linear PDE that in general is impossible to solve analytically. Thereby, numerical methods must be employed to solve it. Two different methods were used. The first one being the explicit-Euler finite difference method, discussed in section 2.2.1. This is used to solve the second task regarding the boundary value control. This is mainly because it simplifies the process of posing the problem as an optimization problem, because it is easy to implement and although it is not as efficient as the other method implemented, it performs sufficiently well in relation to the amount and magnitude of the simulations that were carried out.

The second method used is a spectral method. This was implemented since it aids in computational effort, which was important as a lot of simulations had to be completed. This is discussed further section 2.2.2.

2.2.1 Finite Difference Method

Finite Difference methods are a class of techniques used for solving PDE's. In this paper the specific algorithm of choice is the explicit-Euler method, which can be implemented to solve (1) numerically. All derivatives are approximated by finite differences and the time and space domain are discretized uniformly as follows

$$\text{Space domain} = \Omega = [0, L_x] \times [0, L_y], \qquad \text{Time domain} = \Psi = [0, T],$$

$$0 = x_0, ..., x_{N-1} = L_x, \qquad 0 = y_0, ..., y_{M-1} = L_y, \qquad 0 = t_0, ..., t_{F-1} = T,$$

with

$$\Delta x = x_{i+1} - x_i = \Delta y = y_{j+1} - y_j = h, \qquad \forall i \in \{0, ..., N-2\}, \quad \forall j \in \{0, ..., M-2\}$$
$$\Delta t = t_{n+1} - t_n = k, \qquad \forall n \in \{0, ..., F-2\}.$$

where $u^n_{i,j}$, $v^n_{i,j}$ represent the numerical approximations of $u(x_i, y_j, t_n)$ and $v(x_i, y_j, t_n)$, respectively. N, M will henceforth also be referred to as numΔx, numΔy respectively. Using the finite difference approximations for the derivatives

$$\frac{\partial u}{\partial t} = \frac{u^{n+1}_{i,j} - u^n_{i,j}}{k}$$
$$\nabla^2 u = \frac{u^n_{i+1,j} + u^n_{i-1,j} + u^n_{i,j+1} + u^n_{i,j-1} - 4u^n_{i,j}}{h^2}$$

the explicit-Euler method transforms (1) into

$$u^{n+1}_{i,j} = u^n_{i,j} + \frac{D_u k}{h^2}(u^n_{i+1,j} + u^n_{i-1,j} + u^n_{i,j+1} + u^n_{i,j-1} - 4u^n_{i,j}) + kf(u^n_{i,j}, v^n_{i,j})$$

$$v^{n+1}_{i,j} = v^n_{i,j} + \frac{D_v k}{h^2}(v^n_{i+1,j} + v^n_{i-1,j} + v^n_{i,j+1} + v^n_{i,j-1} - 4v^n_{i,j}) + kg(u^n_{i,j}, v^n_{i,j})$$

Choosing boundary conditions and coupled with the perturbed homogeneous initial states for u and v defined by (4), these sets of equations are solved for all time steps.

2.2.2 Spectral Method

The spectral method is a technique for solving PDE's by instead writing the solution as a sum of basis functions, the basis functions being those that make up a Fourier series, then utilizing Fourier transforms to find the explicit solution. To do this numerically, this is done in a discretized manner, with time and space domain discretized as in 2.2.1. There are a couple of the advantages of posing the problem like this. The first one being the ease of differentiation in the frequency domain compared to the time domain. Moreover, the error decreases exponentially with finer discretization [10] and since the fast Fourier transform (FFT) can be used, the computational effort is decreased drastically.

Again, the equation of interest is (1), which breaks apart into

$$\frac{\partial u}{\partial t} = D_u\Big(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}\Big) + f(u, v) \tag{6}$$

$$\frac{\partial v}{\partial t} = D_v\Big(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2}\Big) + g(u, v) \tag{7}$$

It is easier beginning by focusing on $u(x, y, t)$, omitting $f(u, v)$ and D_u. The solution to the above equation is represented in terms of its orthonormal basis functions $\phi_n(x)\psi_m(y)$.

$$u(x, y, t) = \sum_n \sum_m \hat{u}_{nm}(t)\phi_n(x)\psi_m(y),$$
$$\phi_n(x) = e^{2\pi i n x/L_x}, \quad \psi_m(y) = e^{2\pi i m y/L_y},$$

where $\phi_n(x)$ and $\psi_m(y)$ are the complex harmonic basis functions. Since ϕ_n, ψ_m are periodic they imply periodic boundary conditions on the solution. A discretized approximation of u is sought, where only N of $\phi_n(x)$ and M of $\psi_m(y)$ are included. Hence, making the discretized ansatz for the approximation of u [10]

$$u(x, y, t) = \sum_{n=-N/2+1}^{N/2} \sum_{m=-M/2+1}^{M/2} \hat{u}_{nm}(t) e^{2\pi i n x/L_x} e^{2\pi i m y/L_y}. \tag{8}$$

This describes the discrete inverse Fourier transform in 2 dimensions. The truncation of the Fourier series will obviously lead to an error, but the error decreases exponentially with N so long as $u(x, y, t)$ is smooth, which is the case. The discrete Fourier transform and its inverse are defined as follows

$$u_{jk} = F^{-1}\{\hat{u}_{nm}\} = \sum_{n=-N/2+1}^{N/2} \sum_{m=-M/2+1}^{M/2} \hat{u}_{nm} e^{2\pi i n j/N} e^{2\pi i m k/M} \tag{9}$$

$$\hat{u}_{nm} = F\{u_{jk}\} = \frac{1}{NM} \sum_{j=0}^{N-1} \sum_{k=0}^{M-1} u_{jk} e^{-2\pi i n j/N} e^{-2\pi i m k/M} \tag{10}$$

Inputting equation (8) into (6) yields

$$\frac{\partial u}{\partial t} = \sum_{n=-N/2+1}^{N/2} \sum_{m=-M/2+1}^{M/2} \left[\left(\frac{2\pi i n}{L_x} \right)^2 + \left(\frac{2\pi i m}{L_y} \right)^2 \right] \hat{u}_{nm} e^{2\pi i n x/L_x} e^{2\pi i m y/L_y},$$

and its discretized counterpart

$$\frac{\partial u_{jk}}{\partial t} = \sum_{n=-N/2+1}^{N/2} \sum_{m=-M/2+1}^{M/2} \left[\left(\frac{2\pi i n}{L_x} \right)^2 + \left(\frac{2\pi i m}{L_y} \right)^2 \right] \hat{u}_{nm} e^{2\pi i n j/N} e^{2\pi i m k/M}.$$

Now, substituting $\hat{u}_{nm}$ with (10)

$$\frac{\partial u_{jk}}{\partial t} = \sum_{n=-N/2+1}^{N/2} \sum_{m=-M/2+1}^{M/2} \left[\left(\frac{2\pi i n}{L_x} \right)^2 + \left(\frac{2\pi i m}{L_y} \right)^2 \right] F\{u_{jk}\} e^{2\pi i n j/N} e^{2\pi i m k/M},$$

and identifying the right-hand side with the inverse Fourier transform (9), the expression is simplified to

$$\frac{\partial u_{jk}}{\partial t} = F^{-1}\Big\{ \left[\left(\frac{2\pi i n}{L_x} \right)^2 + \left(\frac{2\pi i m}{L_y} \right)^2 \right] F\{u_{jk}\} \Big\}.$$

Finally, adding $f(u, v)$ and D_u, the final, discretized equation reads

$$\frac{\partial u_{jk}}{\partial t} = D_u F^{-1}\Big\{ \left[\left(\frac{2\pi i n}{L_x} \right)^2 + \left(\frac{2\pi i m}{L_y} \right)^2 \right] F\{u_{jk}\} \Big\} + f(u_{jk}, v_{jk}). \tag{11}$$

Analogously, the equation for v reads

$$\frac{\partial v_{jk}}{\partial t} = D_v F^{-1}\Big\{ \left[\left(\frac{2\pi i n}{L_x} \right)^2 + \left(\frac{2\pi i m}{L_y} \right)^2 \right] F\{v_{jk}\} \Big\} + g(u_{jk}, v_{jk}) \tag{12}$$

Again, using the perturbed homogeneous initial states for u and v described by (4). The system of ordinary differential equations (11) and (12) can then be solved over the discretized time domain.

This procedure is implemented in Python, using `fft2`, `ifft2` (numpy package) to compute the 2-dimensional discrete Fourier transforms by means of FFT and `odeint` (scipy package) to solve the system of ODE's.

2.3 Boundary Value Control

This report studies the properties of boundary control of the Turing PDE. Boundary control being the control of a PDE through its boundary conditions. Specifically, Dirichlet boundary conditions are chosen.

Turing's PDE can, as noted in section 2.1 be split up into two parts: a diffusion and a reaction part

$$\frac{\partial \mathbf{u}}{\partial t} = \underbrace{\mathbf{D}\nabla^2\mathbf{u}}_{\text{diffusion}} + \underbrace{\mathbf{f}(\mathbf{u})}_{\text{reaction}}, \quad \mathbf{f}(\mathbf{u}) = \begin{bmatrix} f(u,v) \\ g(u,v) \end{bmatrix}, \quad \mathbf{u} = \begin{bmatrix} u \\ v \end{bmatrix}, \quad \mathbf{D} = \begin{bmatrix} D_u & 0 \\ 0 & D_v \end{bmatrix} \tag{13}$$

$$\mathbf{u}(x,y,t) = \mathbf{X}(t), \quad (x,y) \in \partial\Omega$$

where $\mathbf{X}(t)$ is the external control function (boundary value control function). The reaction part is linearized for simplification purposes and the resulting equation is discretized using the explicit-Euler finite difference method, described in 2.2.1, yielding equations

$$\frac{du_{i,j}}{dt} = \underbrace{au_{i,j} + bv_{i,j}}_{\text{reaction}} + \underbrace{p\,[u_{i,j} + u_{i-1,j} + u_{i,j-1} + u_{i,j+1} - 4u_{i,j}]}_{\text{diffusion}}, \tag{14}$$

$$\frac{dv_{i,j}}{dt} = \underbrace{cu_{i,j} + dv_{i,j}}_{\text{reaction}} + \underbrace{q\,[v_{i,j} + v_{i-1,j} + v_{i,j-1} + v_{i,j+1} - 4v_{i,j}]}_{\text{diffusion}}. \tag{15}$$

Using the same discretization as in section 2.2.1, the outermost grid points are defined as the boundary. In defining the control problem, the concentration values of these points are gathered in a discretized control function

$$\mathbf{X}(t_n) = \begin{pmatrix} X^u(t_n) \\ X^v(t_n) \end{pmatrix},$$

where $X^u(t_n)$ and $X^v(t_n)$ are column vectors containing the boundary cells of control in the domain of morphogen u and v respectively, at time t_n. Figure 4 depicts the location of the control variables in a 5×5 grid.

	X_1^u	X_2^u	X_3^u	
X_{12}^u	$u_{2,2}$	$u_{2,3}$	$u_{2,4}$	X_4^u
X_{11}^u	$u_{3,2}$	$u_{3,3}$	$u_{3,4}$	X_5^u
X_{10}^u	$u_{4,2}$	$u_{4,3}$	$u_{4,4}$	X_6^u
	X_9^u	X_8^u	X_7^u	

	X_1^v	X_2^v	X_3^v	
X_{12}^v	$v_{2,2}$	$v_{2,3}$	$v_{2,4}$	X_4^v
X_{11}^v	$v_{3,2}$	$v_{3,3}$	$v_{3,4}$	X_5^v
X_{10}^v	$v_{4,2}$	$v_{4,3}$	$v_{4,4}$	X_6^v
	X_9^v	X_8^v	X_7^v	

Figure 4: Depiction of the control points X_i^u, X_i^v in a 5x5 grid.

Having defined a discrete control function $\mathbf{X}(t_n)$, equations (14) and (15) can be written as

$$\begin{pmatrix} u(t_{n+1}) \\ v(t_{n+1}) \end{pmatrix} = \mathbf{A} \begin{pmatrix} u(t_n) \\ v(t_n) \end{pmatrix} + \mathbf{B}\mathbf{X}(t_n).$$

A initial and final pattern of the morphogens u, v are contstruced, with the initial and final pattern denoted as

$$\begin{pmatrix} u_0 \\ v_0 \end{pmatrix}$$

and

$$\begin{pmatrix} u_T \\ v_T \end{pmatrix}$$

respectively. The task is to find $\mathbf{X}(t_n) = \begin{pmatrix} X^u(t_n) \\ X^v(t_n) \end{pmatrix}$ for all time steps t_n, $n \in \{1, \ldots, F-1\}$ that yield the final pattern in a desired time T. This requirement on the $\mathbf{X}(t_n)$'s can further be transformed into

$$\begin{pmatrix} \mathbf{A}^{F-1}\mathbf{B} & \mathbf{A}^{F-2}\mathbf{B} & \ldots & \mathbf{AB} & \mathbf{B} \end{pmatrix} \begin{pmatrix} \mathbf{X}(t_{F-1}) \\ \vdots \\ \mathbf{X}(t_0) \end{pmatrix} = \begin{pmatrix} u_T \\ v_T \end{pmatrix} - \mathbf{A}^F \begin{pmatrix} u_0 \\ v_0 \end{pmatrix} \tag{16}$$

which is the representation used in the implementation of the problem.

Specifically, this report studies so-called minimal energy control. Theoretically, if at least one line of the boundary of the domain Ω is controlled, there are an infinite number of choices of $\mathbf{X}(t)$ to generate any final pattern from any initial pattern. Introducing the cost function

$$\int_0^T ||\mathbf{X}(t)||^2 dt$$

with its discretized counterpart

$$\sum_{n=0}^{F-1} ||\mathbf{X}(t_n)||^2 \Delta t. \tag{17}$$

one arrives at an optimization problem of finding a choice of $\mathbf{X}(t_n)$'s which minimizes its value. The optimization problem can be formulated as

$$\begin{aligned} &\min_{\mathbf{X}(t_n),\, n\in\{0,\ldots,F-1\}} \sum_{n=0}^{F-1} ||\mathbf{X}(t_n)||^2 \Delta t \\ \text{s.t. } &\begin{pmatrix} u(t_{n+1}) \\ v(t_{n+1}) \end{pmatrix} = \mathbf{A} \begin{pmatrix} u(t_n) \\ v(t_n) \end{pmatrix} + \mathbf{B}\mathbf{X}(t_n) \\ \text{with } &\mathbf{u}(0) = \begin{pmatrix} u_0 \\ v_0 \end{pmatrix},\ \mathbf{u}(t_F) = \begin{pmatrix} u_T \\ v_T \end{pmatrix} \text{ given.} \end{aligned}$$

MATLAB is used to numerically find a solution to the optimization problem, finding $\mathbf{X}(t_n)$'s adhering to the constraint (16) whilst minimizing (17), using the nonlinear progamming solver `fmincon`.

3 Results and Discussion

3.1 Correctness of Implementation of Spectral Method

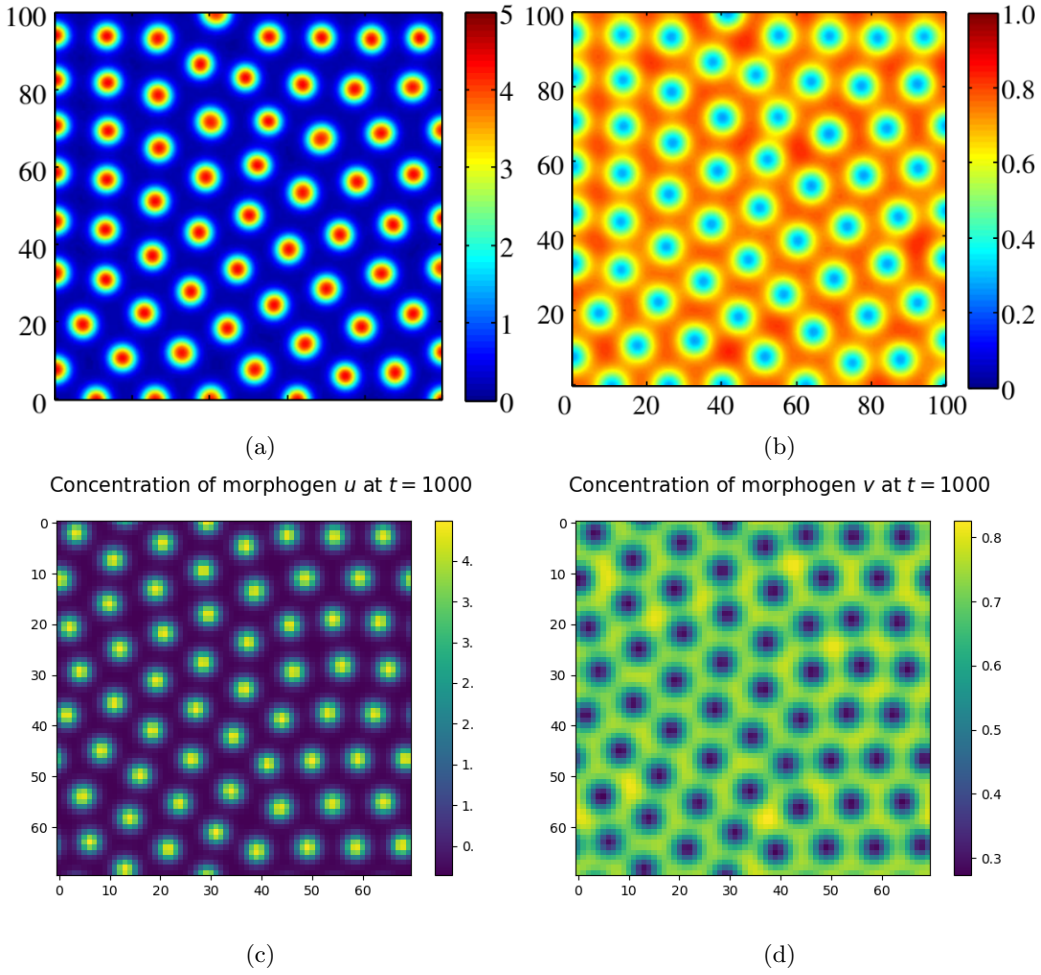


Figure 5: u and v after 1000 time steps using Schnakenberg kinetics. Parameters: $c_{-1} = 1$, $c_1 = 0.1$, $c_2 = 0.9$, $c_3 = 1$. (a) and (b) are known results from [5]. (c) and (d) were generated by our simulations.

Using the spectral method as described in section 2.2.2 a simulation was run with identical parameters as that of a simulation from article [5]. The model parameters are tabulated in table 1 with additional simulation parameters $t_{end} = 1000$, $L = 100$ and $\text{num}\Delta x = \text{num}\Delta y = 70$. The pattern at time $t = 1000$ in our simulation and the acrticle's are depicted in figure 5. Figure 5a, 5b are the known result from the article and 5c, 5d are our simulation results. When comparing the aforementioned figures it is clearly visible that our implementation and method arrives at similar results as that of [5] despite the fact that different boundary conditions were used. Periodic in our case and zero-flux in the article. Thereby supporting the aptness of our results.

parameter	c_{-1}	c_1	c_2	c_3	D_u	D_v
value	1	0.1	0.9	1	1	40

Table 1: Parameters of the Schnakenberg model from figure (1) in article [5].

3.2 Convergence

Continuing the analysis, studies were made into the convergence rate of the system with respect to time. Apart from the long Scnakenberg simulation mentioned in section 3.1, two additional long simulation (referred to as GM1 and GM2) with instead the Grier-Meinhardt kinetics and parameters as in table 2 were carried out.

	GM1	GM2
parameter	value	
c_1	0	0
c_2	0.5	0.5
c_3	0.5	1.15
c_4	0.5	0.5
c_5	0.45	0.45
D_u	0.1	0.1
D_v	2	2
k	0.238	0.081
t_{end}	1500	1000
L	100	70
num Δx	80	70

Table 2: Parameters used in long simulations with the Grier-Meinhardt kinetics.

Figure 6 shows the concentrations of morphogens u and v at some different time steps for the Schankenberg and GM1 simulation and figure 7 shows plots of $\sum |u_i - u_{i+1}|^2$ and $\sum |v_i - v_{i+1}|^2$ for time steps $i = 1, \ldots, t_{F-1}$ in all three simulations. u_i, v_i denote the concentration matrices at time step i, subtraction and exponentiation is done element wise, and the sum is taken over all elements in the matrix. As visible in figures 6, 7 the system converges toward some stable configuration. As t increases, the change of concentration between time step decreases roughly polynomially after $t = 200$. When looking at figures 6, 7 the conclusion can be drawn that the prominent features of the pattern, stemming from a set of parameters, becomes clear after about 200 time steps. It is however not clear if this behaviour persists for all configurations of parameters. For our continued analysis however, simulations will be run until about $t = 200$, with some variation depending on the set of parameters, if necessary for drawing conclusions about the properties of the resulting pattern.

3.3 Gridsize

Some analysis was also conducted regarding an appropriate choice of discretization for simulations of the Turing PDE. Figure 8 depicts simulations results of the Schnakenberg model and the Grier-Meinhardt model. A plethora of simulations were performed, where grid sizes were varied with a fix domain size $L_x = L_y = 30$ and for different sets of parameters. As is noticeable, too large choices of grid size results in a non-homogeneous spatiotemporally stable pattern being unable to form, i.e finite size errors disrupting the dynamics of the Turing PDE, resulting in the system converging towards a homogeneous distribution of concentration. The grid size for which these errors seem to have a particularly pronounced effect, appear to depend on the size of the features (for example spots or snakes) of the Turing pattern, which in turn depend on the parameter set

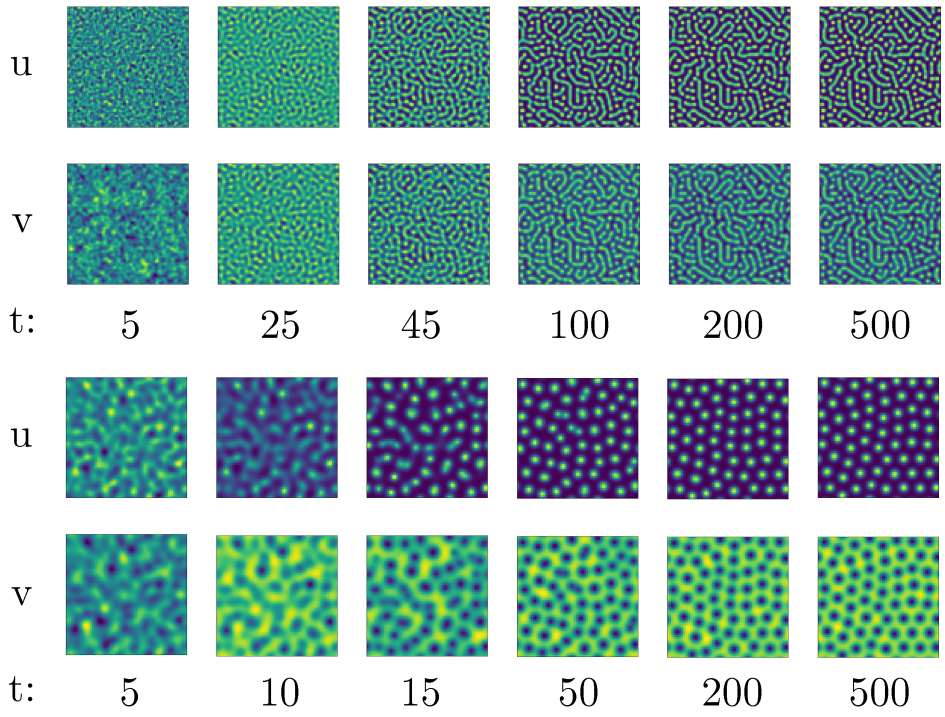


Figure 6: Concentrations of morphogens u, v (shown to the left) at some different time steps (integers under the pattern pictures) for two of the long simulations. The GM1 Grier-Meinhardt simulation with parameters as in table 2 in the upper part of the figure, and the Schnakenberg simulation in the bottom part.

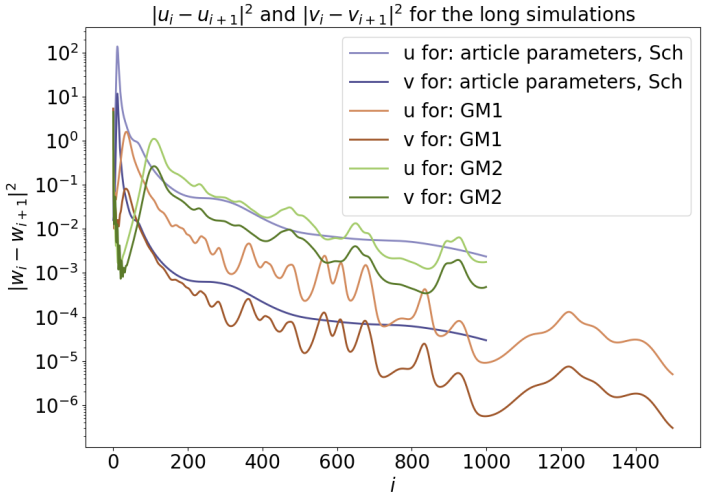


Figure 7: Plots of $\sum |u_i - u_{i+1}|^2$ and $\sum |v_i - v_{i+1}|^2$ for all time steps in the long simulations.

(which is discussed below). For our continued analysis, grid sizes will be chosen small enough as to not noticeably disrupt the PDE dynamics whilst keeping the run time of our simulations reasonably low.

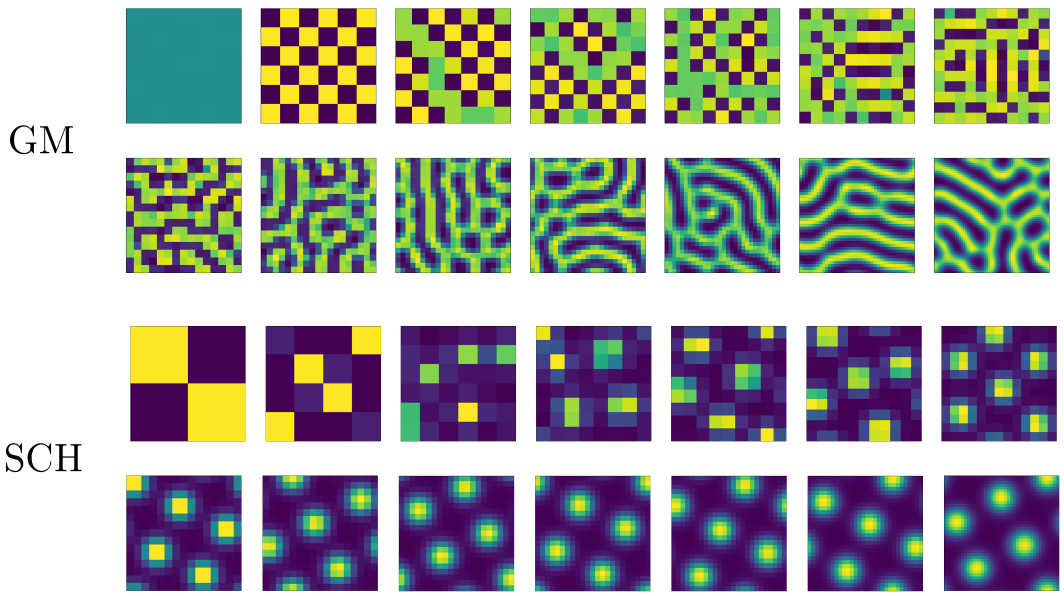


Figure 8: Patterns generated with varying grid sizes using Grier-Mienhardt (top) and Schnakenberg (bottom) kinetics.

3.4 Grier-Meinhardt

Firstly, the characteristics of the patterns evolving from the Grier-Meinhardt model and its parameters were studied. Simulations were performed to investigate the effect which the different parameters has on the non-homogeneous diffusion driven pattern. Figures 9, 10 shows the patterns at the last time step $t_{end} = 250$ for a collection of simulations where one parameter at a time was varied from the base set of parameters in table 3. The images in figures 9, 10 are depictions of the concentration of morphogens u, v respectively, from corresponding simulations. The depicted simulations are a selection from a larger set of simulations, where simulations of particular visual interest were chosen. The general changes in appearance of the patterns have been compiled in table 4.

parameter	D_u	D_v	c_1	c_2	c_3	c_4	c_5	k
value	0.1	2.0	0.0	0.5	0.5	0.5	0.45	0.081

Table 3: Base parameter values for studies of the Grier-Meinhardt kinetics.

This dynamic between the two substances, coupled with the complex dynamics of the system give rise to the patterns in figures 9, 10. Upon varying a parameter in a permissible interval (in terms of the DDI criteria), the pattern changes appearance due to the dynamics of the system changing. The most general conclusion that can be drawn on the appearance in relation to parameter values is that certain values entail more connected structures (such as honeycombs or snake-like patterns as seen in table 4), while others lead to a more scattered patterns (such as spots/rods).

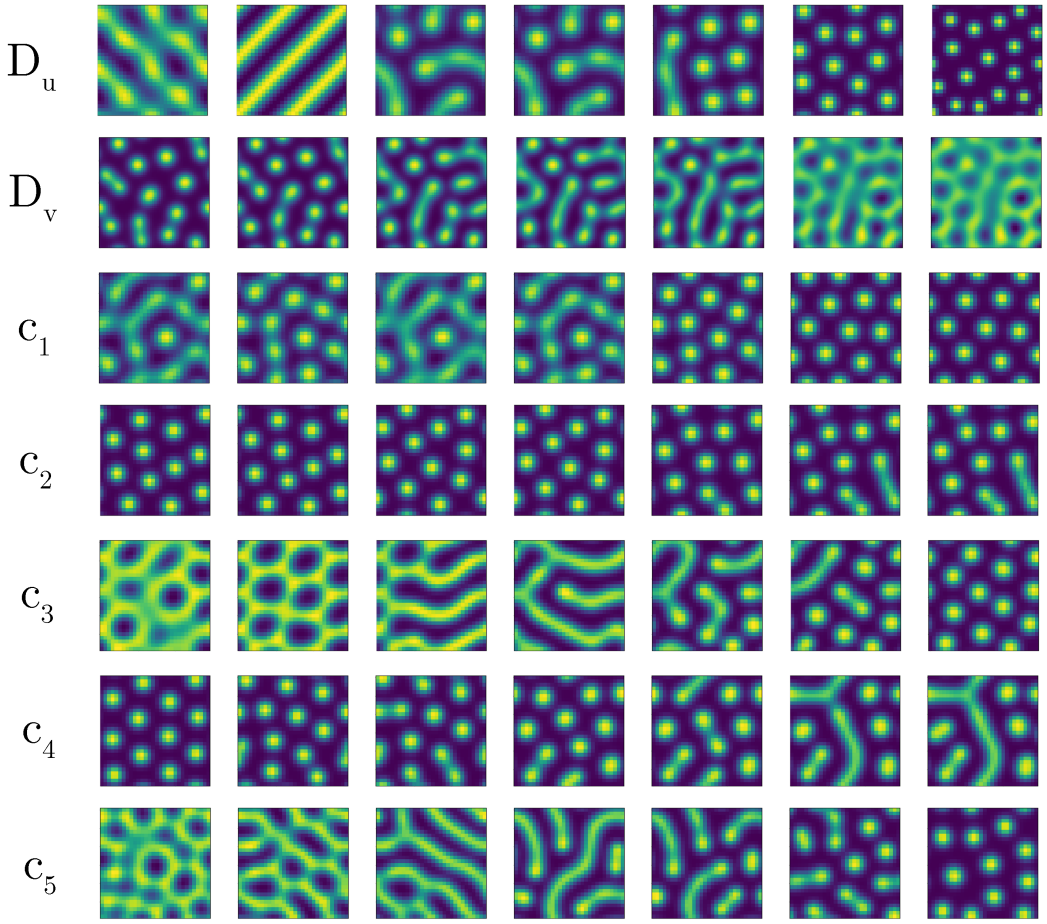


Figure 9: Simulations of u with the Grier-Meinhardt kinetics, where one parameter at a time was varied (specified to the left) from the set in table 3. Images are sorted in descending parameter values to the right.

behaviour	decrease in value	parameter	increase in value	behaviour
stripes/tendrils	$\leftarrow$	D_u	$\rightarrow$	rods/spots
spots/rods	$\leftarrow$	D_v	$\rightarrow$	tight honeycomb
snake-like	$\leftarrow$	c_1	$\rightarrow$	spots
spots	$\leftarrow$	c_2	$\rightarrow$	rods/spots
honeycomb/snake-like	$\leftarrow$	c_3	$\rightarrow$	tendrils/rods/spots
spots	$\leftarrow$	c_4	$\rightarrow$	tendrils/rods/spots
tight honeycomb	$\leftarrow$	c_5	$\rightarrow$	tendrils/rods/spots

Table 4: Compilation of the effect that the parameters of the Grier-Meinhardt reaction function has on the appearance of the generated Turing pattern.

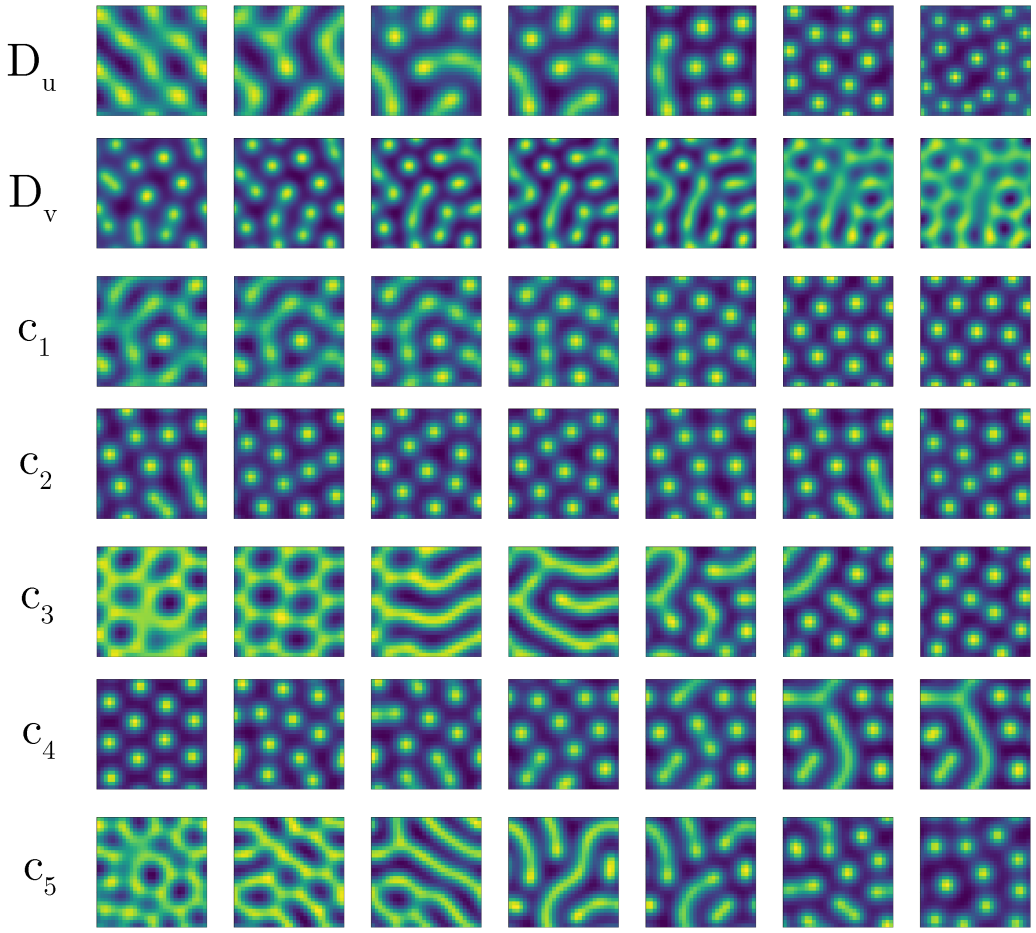


Figure 10: The corresponding simulations of figure 9 but instead showing the concentration of morphogen v. For further details see figure 9.

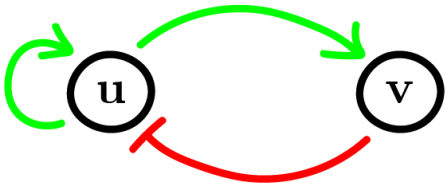


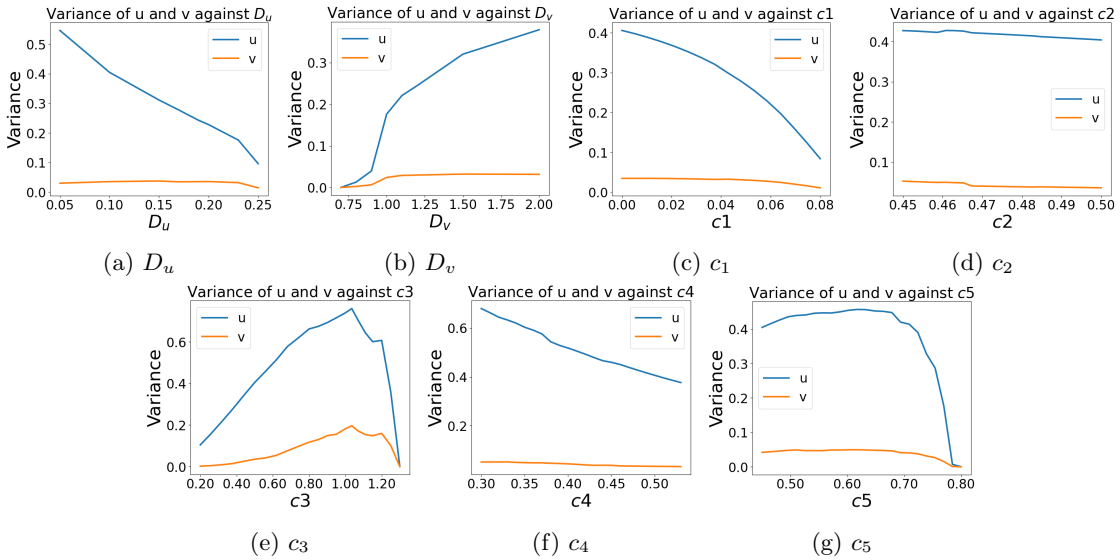
Figure 11: Reaction topology visualization of the Grier-Meinhardt reaction kinetics.

Morphogens u and v act as activator and inhibitor, respectively, as discussed in section 2.1. Figure 11 shows a visualisation of the reaction topology of the Grier-Meinhardt activator inhibitor model. The reason as to why the patterns of u and v appear similar in figures 9 and 10 in terms of structure can be explained with this topology in mind. u can produce more of itself, while v cannot, since its synthesis is dependent on u. Therefore, v is more prevalent in regions where there is also a lot of u.

D_v denotes the diffusion rate of the inhibitor. One of the critical conditions for Turing patterns is the long-range activation, short-range inhibition effect discussed in the introduction. Thus, D_v has to be significantly larger than D_u. As D_v is decreased, the ability of v to inhibit u is restricted to taking effect locally, thus allowing u to spread in larger regions, without being scattered. Hence the larger honeycomb-like pattern to the right in the D_v row of figure 9. This is supported by the fact that the nature of the evolution of u is flipped for D_u, since the ratio D_v/D_u is increased with decreasing D_u, thus exaggerating the short-range activation/long-range inhibition effect (see the D_u row in figure 9).

Figure 12 depicts the variance of both morhpogen concentrations for different choices of parameter values. The variance is related to how much the peaks and troughs of the morphogen concentrations vary in magnitude. Thus, a lower variance means the patterns are closer to a homogeneous state. Since the color range of the pictures in figures 9, 10 is fitted to the range of values of the concentration, variance plots gives us further insight about the patterns. From figure 12b it is deduced that the higher the value of D_v, the more pronounced the features in the corresponding images in figure 9 are. So, the long-range inhibiting effect of v through D_v tends to accumulate u in more distinct agglomerates. For lower D_v, where this effect is not as pronounced, the pattern of u is more connected but not as conspicuous. The effect of increasing/decreasing D_u, D_v are mirrored, as was the case for the characteristics of patterns. This, again, owes to the ratio between the two being of essence.

It is interesting to note that if D_u, D_v are varied to a point where the ratio D_v/D_u is insufficiently small, the parameter set fall outside of the DDI parameter space and the system converges to a homogeneous state. The variance of D_v does not exhibit linear behavior, instead it drops off to zero swiftly after a threshold. The value for which it becomes zero coincides with it falling out of the permissible parameter space.



(a) D_u (b) D_v (c) c_1 (d) c_2

(e) c_3 (f) c_4 (g) c_5

Figure 12: Variances of morphogen concentration against the varied parameters, corresponding to the images in figures 9 and 10.

This entire discussion goes to show that the dynamics of this non-linear system is quite complex. To further analyse the specifics of the nature of the generated patterns, such as why certain parameter values tend to generate longer tendril-like structures, while others yield spots, would perhaps require a more in-depth analysis of frequencies and modes of the system dynamics.

3.5 Schnakenberg

Secondly, the features of the patterns evolving from the Schnakenberg model and its parameters were investigated. As in the previous section, parameters were varied one at a time from a base set of parameters, with the base set of parameters given in table 1 (the parameters from article [5]). Figures 13 and 14 show the patterns of u and v, respectively from corresponding simulations. The simulation time was $t_{end} = 250$. Each image corresponds to a unique parameter value. The general changes in appearance of the patterns have been compiled in table 5.

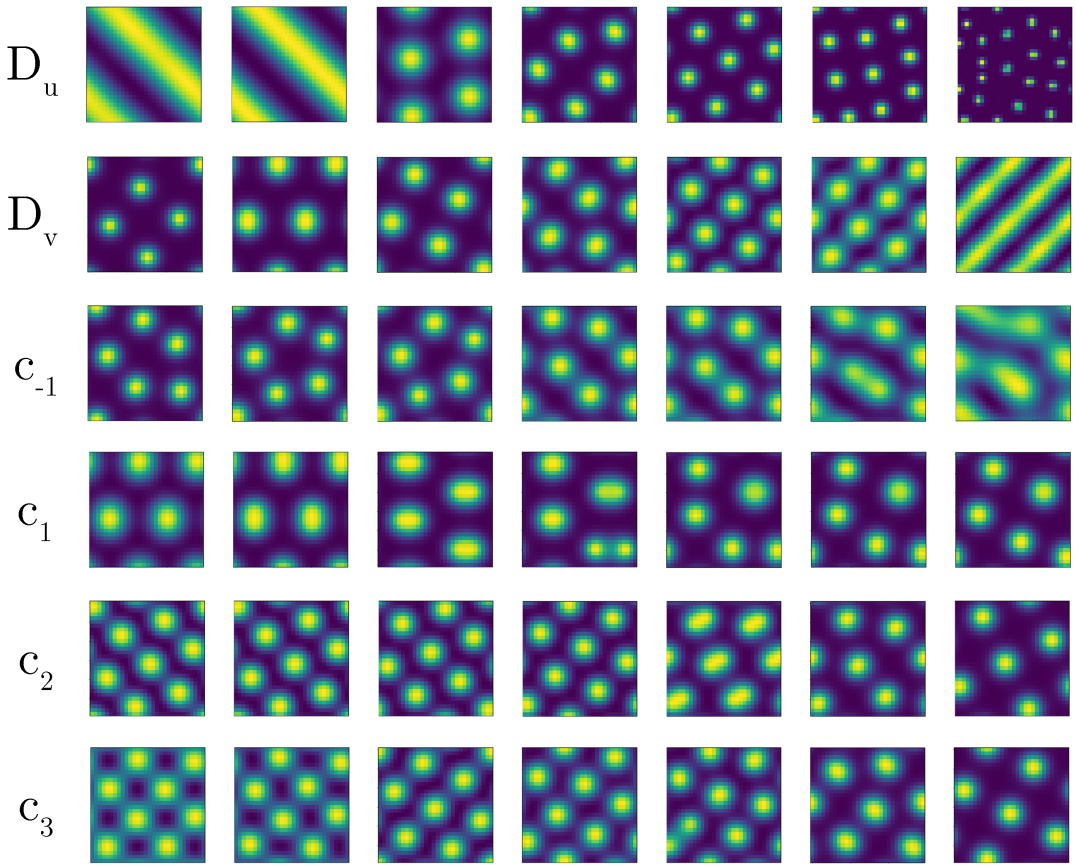


Figure 13: Simulations of the Turing PDE with the Schnakenberg kinetics, showing the concentration of morphogen u, where one parameter at a time was varied (specified to the left) from set in table 1 and with smaller parameter values going to the right.

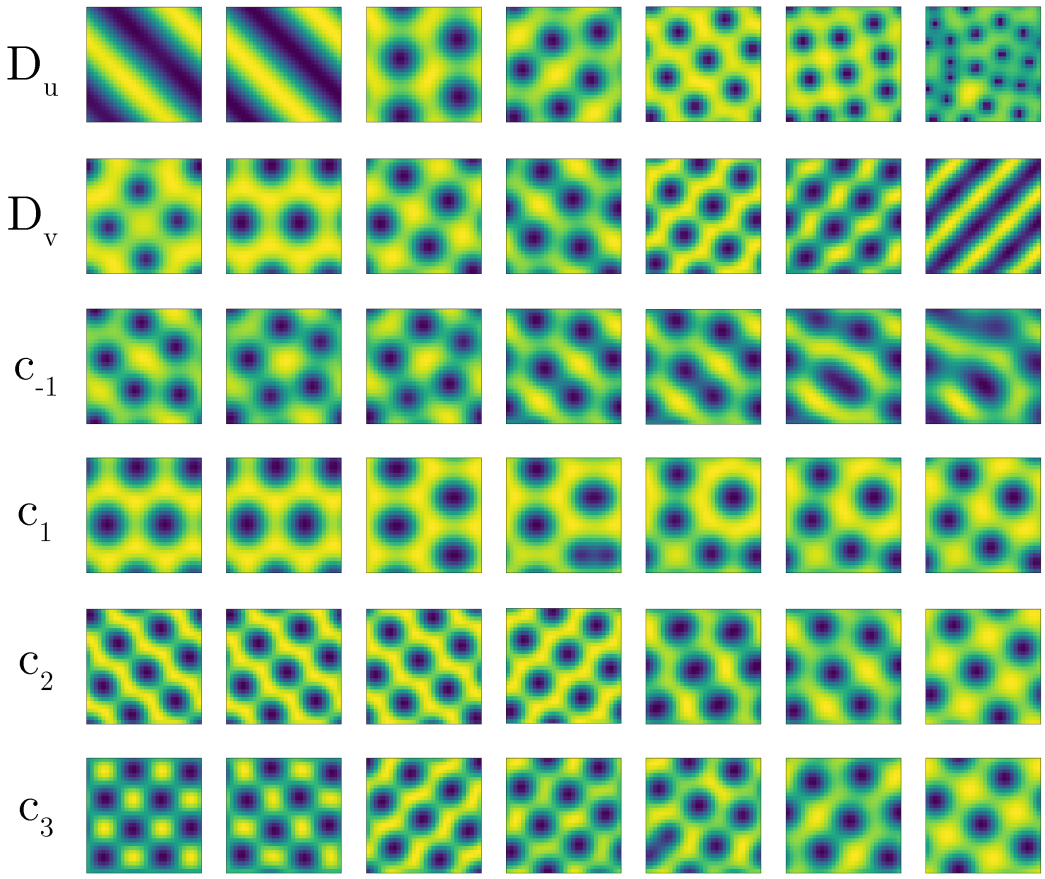


Figure 14: The corresponding simulations of figure 13 but instead showing the concentration of morphogen v. For further details see figure 13.

behaviour	decrease in value	parameter	increase in value	behaviour
stripes	$\leftarrow$	D_u	$\rightarrow$	spots
spots	$\leftarrow$	D_v	$\rightarrow$	stripes
spots	$\leftarrow$	c_{-1}	$\rightarrow$	blotches/tendrils
spots (larger)	$\leftarrow$	c_1	$\rightarrow$	spots (smaller)
aligned spots	$\leftarrow$	c_2	$\rightarrow$	spots
checkerboard	$\leftarrow$	c_3	$\rightarrow$	spots

Table 5: Compilation of the effect which the parameters of the Schnakenberg reaction function has on the appearance of the generated Turing pattern.

Figure 15 depicts the reaction topology of the Schnakenberg substrate-activator model as discussed in section 2.1.

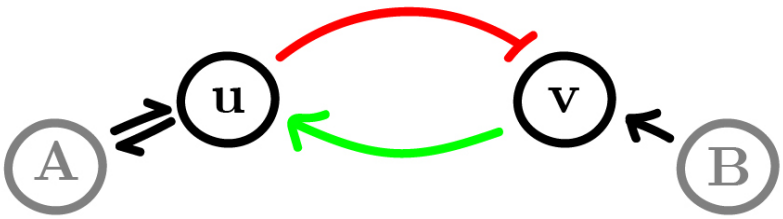
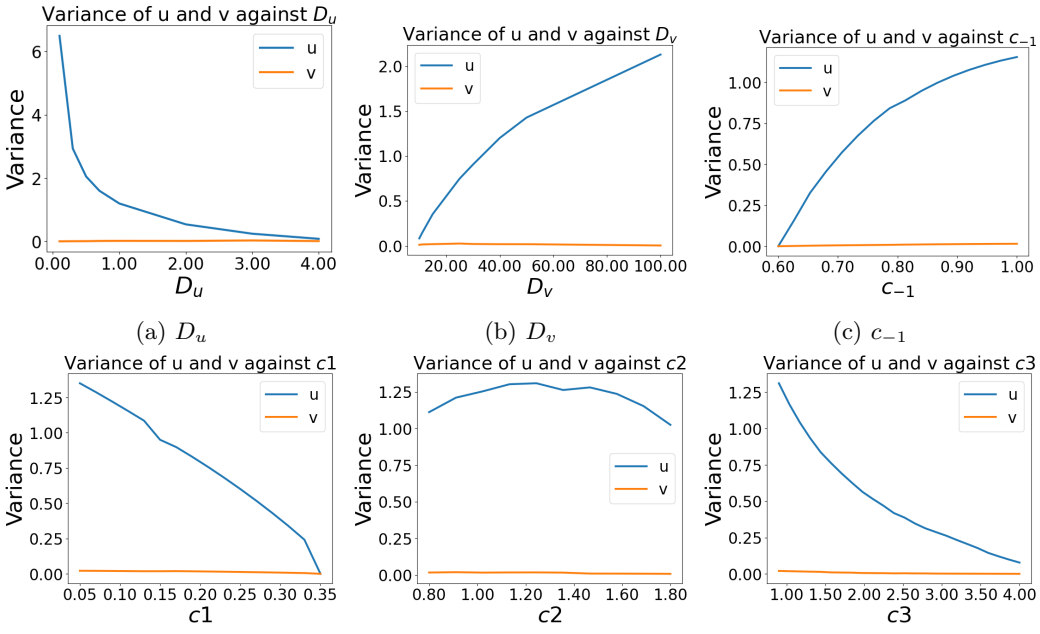


Figure 15: Visualization of the reaction topology of the Schnakenberg kinetics.

This topology is different from that of the Grier-Meinhardt model and thus results in different Turing patterns. Here, the patterns of u and v appear inverted. This is due to u consuming v to produce more of itself, so in regions where both of the morphogens are present, v will tend to be consumed by u.

Shifting the focus to the diffusion coefficients, a large value of D_u makes u spread out faster, enabling it to react with and consume v producing larger areas where the concentration of u is pronounced, as seen to the left on row D_u in figure 13. For lower values, since it is not able to distribute as readily, it will be confined to smaller areas. As was the case for the Grier-Meinhardt kinetics, the inverse relation is true for D_v, again showing the significance of the ratio between the two.

Turning the attention to figure 16, depicting the variances of the morphogens in the patterns, it is evident that the patterns whose behavior are categorized as "spots" in table 5 have higher variance and are thus more distinct, than those on the other side of the behavioral spectrum. The exception being c_2 where there seems to be a trade-off between number of spots and spot morphogen concentration, leading to a relatively constant variance for different parameter values.



(a) D_u (b) D_v (c) c_{-1}

(d) c_1 (e) c_2 (f) c_3

Figure 16: Variances of morphogen concentration against the varied parameters, corresponding to the images in figures 13 and 14.

In general, the Schnakenberg model does not seem to be able to render as complex and varied of a pattern spectrum as easily as the Grier-meinhardt model does. There are less parameters and the models of the reaction kinetics differ in complexity. The simplicity on the other hand, makes for a less burdensome process of parameter choice and yet captures the qualities of the Turing pattern model.

3.6 Bundary Value Control

Studies of the properties of boundary value control were performed. With the change of programming software of implementation, came also a change in the visualisation method of results. In this section the concentration of morphogens u, v were visualized using the properties of the RGB color model, with the concentration of morphogen u visualized via the amount of red in the figures and the concentration of morphogen v the amount of green in the pictures, with the effect that occurrence of both morphogen u and v in the same cell resulting in a color tending towards yellow. An analysis was carried out on the effect which the amount of grid points of control has on the final value of the cost function when solving the optimization problem outlined in section 2.3. Tests were performed by varying the amount of points on the boundary of the defined space domain which are optimized and letting the other boundary cells assume a value of zero. In figure 18 the cost of the control function which minimizes the cost has been plotted against the number of boundary grid points of control. This by starting at an amount of grid points of control, which successfully solved the optimization problem given the parameters in table 6

parameter	a	b	c	d	p	q
value	2	1	1	1	1	1

Table 6: Parameter values used when studying the relation between amount of grid points of control and cost, in the minimal energy optimization problem.

and incrementally increasing the amount of boundary points of control. Additional paramaters for simulation were: $\text{num}\Delta x = \text{num}\Delta y = 20$, $t_F = 5$. All optimizations used the same initial pattern, namely the pattern in which the concentration of u and v is zero at all points, and some different final patterns where tested, as depicted in figure 17.

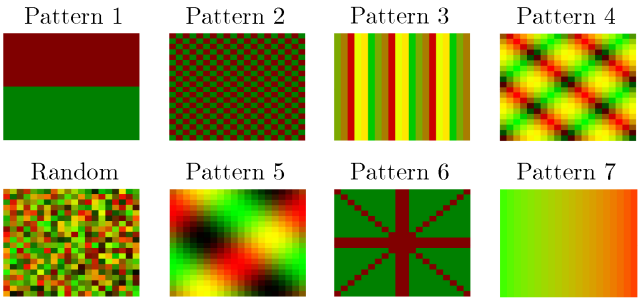


Figure 17: Some of the patterns used when studying the relation between amount of grid points of control and cost, in the minimal energy optimization problem. The RGB value of red denotes the concentration of u and green the concentration of v.

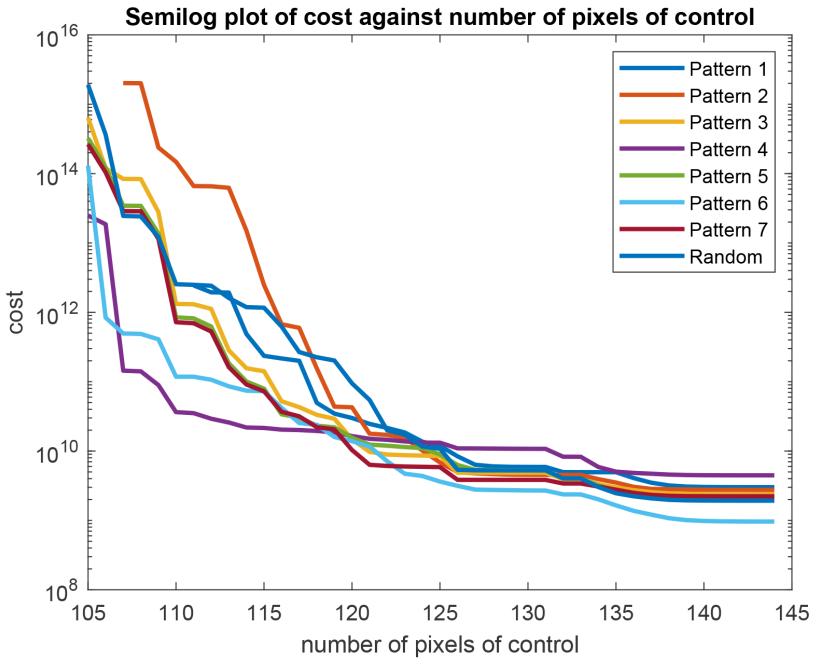


Figure 18: Plot of value of the cost function for the minimized control function against number of pixels of control.

Figure 18 shows that the minimal cost for reaching a desired pattern via boundary value control decreases drastically for an increased amount of points of control. This is intuitive since the degrees of freedom for the minimization increases as the amount of grid points increases. As discussed in 2.3, theoretically, in the case of the continuous formulation of boundary control, equation (13), one only needs to control one line of the boundary of the domain Ω to reach any desired pattern. This however seemed not to be the case for our simulations. As discernible in figure 18 the minimum amount of pixels of control studied was 105, when in fact one line of control would correspond to only 18 pixels of control. Our implementation of the minimization problem couldn't find any choice of the $\mathbf{X}(t_n)$'s, at least within a reasonable run time, such that the constraint of equation (16) was sufficiently satisfied for amounts of points of control below ~ 105. The reason as to why our implementation didn't correspond with the theory could likely be credited to the discretization of the model or the choice of t_F in proportion to the diffusion coefficients D_u, D_v. The latter, since an insufficiently small choice of t_F in relation to the diffusion coefficients could lead to the effects of values of the concentration at the boundary not being able to reach all grid points sufficiently within the given time frame. These effects could be further studied, but our analysis was enough to establish some general behaviour between the amount of grid points and the cost of the reached minimized control function.

4 Conclusions

To summarize, Turing's model exhibits great complexity and it is difficult to draw valuable conclusions about the nature of the generated patterns solely on the basis of parameter choice. No real explanation was found as to why specific structures (like honeycomb patterns) emerged, however some rough trends could be noticed. The difference in topology between the two reaction models had the most pronounced effect, Grier-Meinhardt and Schnakenberg giving rise to vastly dissimilar relationships between u and v in the generated patterns. In general, the Schnakenberg model did not seem to be able to render as complex and varied of a pattern spectrum as easily as the Grier-meinhardt model did. Specifically, the patterns generated by the Grier-Meinhardt model exhibited tendril-like patterns and patterns connected to a larger degree than those generated by the Schnakenberg model, which were instead mostly scattered. There are on the other hand less parameters in the latter and the topology of the reaction kinetics differ in complexity. The simplicity makes for a less burdensome process of parameter choice. Studying minimal energy control of the Turing PDE it was concluded that an increase in amount of grid points of control substantially decreases the energy expenditure.

References

[1] Casper H. L. Beentjes. *Pattern Formation Analysis In The Schnakenberg Model*. URL: https://cbeentjes.github.io/files/Ramblings/PatternFormationSchnakenberg.pdf. (accessed: 6/5/2022).

[2] Alfred Grier and Hans Meinhardt. "Applications of a Theory of Biological Pattern Formation Based on Lateral Inhibition". In: *Journal of Cell Science* 15 (2 1974), pp. 321–346. DOI: 10.1242/jcs.15.2.321. URL: https://doi.org/10.1242/jcs.15.2.321.

[3] Alfred Grier and Hans Meinhardt. "Pattern formation by local self-activation and lateral inhibition". In: *BioEssays* 22 (8 2000), pp. 753–760. DOI: 10.1002/1521-1878(200008)22:8<753::AID-BIES9>3.0.CO;2-Z. URL: https://doi.org/10.1002/1521-1878(200008)22:8%3C753::AID-BIES9%3E3.0.CO;2-Z.

[4] G. F. Oster J. D. Murray and A. K. Harris. "A mechanical model for mesenchymal morphogenesis". In: *Journal of Mathematical Biology* 17 (1 1983), pp. 125–129. DOI: 10.1007/BF00276117. URL: https://doi.org/10.1007/BF00276117.

[5] Maini Philip K., Wooley Thomas E., Baker Ruth E., Gaffney Eamonn A., and Lee S. Seirin. "Turing's model for biological pattern formation and the robustness problem". In: *Interface Focus* 2 (4 2012), pp. 487–496. DOI: 10.1098/rsfs.2011.01134. URL: http://doi.org/10.1098/rsfs.2011.0113.

[6] Shigeru Kondo and Takashi Miura. "Reaction-Diffusion Model as a Framework for Understanding Biological Pattern Formation". In: *Science* 329 (5999 2010), pp. 1616–1620. DOI: 10.1126/science.1179047. URL: http://doi.org/10.1126/science.1179047.

[7] P. K. Maini, M. R. Myerscough, K. H. Winters, and J. D. Murray. "Bifurcating spatially heterogeneous solutions in a chemotaxis model for biological pattern generation". In: *Bulletin of Mathematical Biology* 53 (5 1991), pp. 701–719. DOI: 10.1007/BF02461550. URL: https://doi.org/10.1007/BF02461550.

[8] Christian Science Monitor. *Female tapir cub, Parima, jumps through their compound at the Hagenbeck zoo in Hamburg, northern Germany, May 15, 2012.* [Online; accessed May 24, 2022]. URL: https://www.pinterest.com/pin/animal-of-the-day--277393658271234029/.

[9] Eiichi Yoshimoto Motoomi Yamaguchi and Shigeru Kondo. "Pattern regulation in the stripe of zebrafish suggests an underlying dynamic and autonomous mechanism". In: *Applied Physical Sciences* 104 (12 2007), pp. 4790–4793. DOI: 10.1073/pnas.0607790104. URL: https://doi.org/10.1073/pnas.0607790104.

[10] Philipp Schlatter. *Spectral Methods - Computational Fluid Dynamics SG2212*. URL: https://www.mech.kth.se/~ardeshir/courses/literature/Notes_Spectral_Methods.pdf. (accessed: 5/5/2022).

[11] David Schnoerr Sean T. Vittadello Thomas Leyshon and Michael P. H. Stumpf. "Turing pattern design principles and their robustness". In: *Philosophical Transactions A* 379 (2213 2021). DOI: 10.1098/rsta.2020.0272. URL: https://doi.org/10.1098/rsta.2020.0272.

[12] Shutterstock. *Zebra in the grasslands of the Serengeti at dawn in Tanzania, East Africa.* [Online; accessed May 24, 2022]. URL: https://www.livescience.com/27443-zebras.html.

[13] N. V. Swindale. "A model for the formation of ocular dominance stripes". In: *Proceedings of the Royal Society B: Biological Sciences* 208 (1171 1980), pp. 321–346. DOI: 10.1098/rspb.1980.0051. URL: https://doi.org/10.1098/rspb.1980.0051.

[14] Kyrkans Tidning. *Rådjurssäkra blommor*. [Online; accessed May 24, 2022]. URL: https://www.svenskakyrkan.se/partillekyrkogardsforvaltning/radjurssakra-blommor.

[15] WWF. *Cheetah*. [Online; accessed May 24, 2022]. URL: https://www.wwf.se/djur/gepard/#artdata.