CHALMERS
UNIVERSITY OF TECHNOLOGY

The effect of holes on pattern formation in two species two dimensional reaction-diffusion systems

Master's thesis in Engineering Mathematics and Computational Science

SEBASTIAN PERSSON

Department of Mathematical Sciences
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2020

Master's thesis 2020

The effect of holes on pattern formation in two species two dimensional reaction-diffusion systems

SEBASTIAN PERSSON

CHALMERS
UNIVERSITY OF TECHNOLOGY

Department of Mathematical Sciences
Division of Applied Mathematics and Statistics
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2020

The effect of holes on pattern formation in two species two dimensional reaction-diffusion systems

SEBASTIAN PERSSON

Supervisor: Johannes Borgqvist, Department of Mathematical Sciences, Gothenburg University
Examiner: Marija Cvijovic, Department of Mathematical Sciences, Gothenburg University

Master's Thesis 2020
Department of Mathematical Sciences
Division of Applied Mathematics and Statistics
Chalmers University of Technology
SE-412 96 Gothenburg

Typeset in LaTeX
Printed by Department of Mathematical Sciences
Gothenburg, Sweden 2020

The effect of holes on pattern formation in two species two dimensional reaction-diffusion systems
SEBASTIAN PERSSON
Department of Mathematical Sciences
Chalmers University of Technology

Abstract

Spatial patterns arise in a wide range of biological processes ranging from embryonic development to cell polarisation. Owing to the complexity of pattern formation, attempts to understand it often resorts to modelling by reaction-diffusion (RD) models, which is a model type consisting of coupled partial differential equations (PDE:s). In biological systems, it has been observed that some patterns form regions of high concentration (poles) close to regions in which the species relevant for pattern formation cannot enter (holes). In order to understand the impact of holes on RD-models, and if RD-models can capture this behaviour with poles being confined on domains with holes, this thesis investigated two questions; (1) the impact of holes on pattern formation, (2) if poles by some strategy can be spatially confined on a domain containing holes.

In order to answer these questions, two classical models, the Schnakenberg and Gierer-Meinhardt, were simulated on a two dimensional domain with zero, five, seven and 20 small densely packed circular holes. For solving the RD-systems numerically on such domains a finite element method (FEM) was implemented.

The results suggested that on a domain with many holes poles have a tendency to accumulate close to, or directly in, the region dense in holes. The exact reason behind this behaviour is not known, but it might be due to the model species being confined in the diffusion-restricted region between holes. Regarding the control of pole formation, changing parameters outside the Turing space in a sub-region proved efficient for spatially confining poles to a specific region. Although potentially useful for recreating observed patterns, it should be noted that the usage of this method raises the question of why the parameter values are different in a sub-region.

Keywords: Reaction-diffusion, holes, Schnakenberg, Gierer-Meinhardt, FeniCS, Gmsh

Acknowledgements

I would like to offer my special thanks to Marija Cvijovic for inviting me to her group and being my examiner and supervisor. Thank you for your support, and thank you for always taking the time to offer me advice and guidance when I was in the need of it. To my supervisor Johannes Borgqvist, I would like to thank you for your never ending optimism and useful advice and comments. Also, thank you to Barbara, Felix, Linnea, Johannes, Julia, Marija, Niek, Patrik and Svenja for making my time at Cvijovic lab enjoyable.

Lastly, to my friends and family. Thank you for always supporting and believing in me, it means a lot to me!

Sebastian Persson, Gothenburg, January 2020

Contents

List of Figures

1

Introduction

Spatial pattern formation is a common process in biology [1]. For example, patterns arise during fur development on animals like giraffes, during embryonic development of mammals in which patterns regulate the shape of the embryo [1], and during cell polarisation in the bakers yeast *Saccharomyces cerevisiae* [2]. These are just a few examples of patterns in nature, but they highlight that an understanding of pattern formation is required in order to understand processes ranging from embryonic development to single-cell level polarisation.

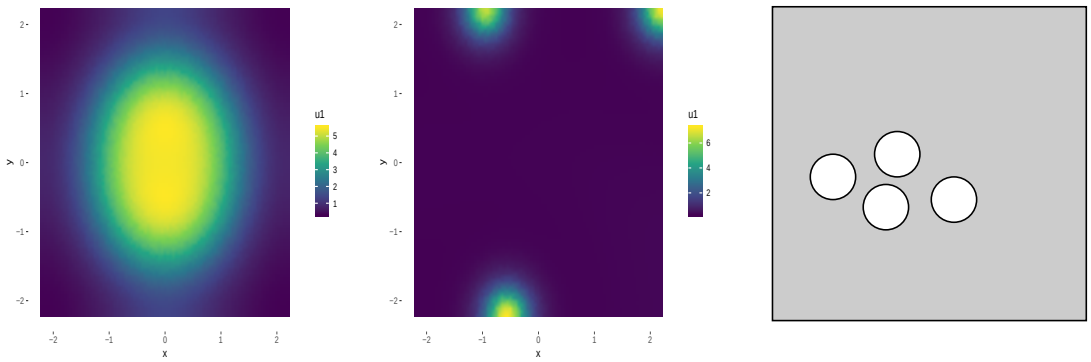
Understanding pattern formation is far from trivial, as the formation of a spatial pattern is a highly complex process [1]. Nonetheless, mathematical modelling has proven useful for understanding and recreating observed patterns [1, 3, 4, 5]. Among the models used to describe pattern formation, the most commonly used and most successful one is the reaction-diffusion (RD) model [4], with the following general form:

$$\frac{\partial \mathbf{c}}{\partial t} = \mathbf{f}(\mathbf{c}) + \mathbf{D}\nabla^2\mathbf{c} \tag{1.1}$$

Here $\mathbf{c}$ is a vector of concentrations (species), $\mathbf{f}$ is a non-linear often bounded reaction term, $\mathbf{D}$ is the diffusivity matrix and ∇^2 is the Laplacian operator. That the system in Eq. (1.1) can give rise to patterns was first demonstrated in a paper from 1952 by Alan Turing, where stable patterns were created using a RD-model with two chemical species [6]. Although producing patterns, Turing's model had a problem with being mechanistically unrealistic. Fortunately, since Turing's time this problem has been addressed and more realistic RD-models have appeared. Among these models, two of the most well studied are the Schnakenberg model [7], Eq. (2.8), and Gierer-Meinhardt model [8], Eq. (2.6) (Fig. 1.1a and 1.1b).

If a pattern is formed on a domain that approximates a flat surface, e.g a butterfly wing, a rectangular domain (Fig. 1.1a) is a reasonable choice for studying said pattern. However, sometimes patterns are formed on more complex domains. An example of such a complex, but biologically plausible, domain is one that contains holes in which the species in the model cannot enter (Fig. 1.1c). These holes can from a biological perspective correspond to budding scars on *S. cerevisiae*[1] [9], or patches of fur where certain molecules cannot diffuse. In the viewpoint of Eq. (1.1) these holes are zero concentration ($\mathbf{c} = \mathbf{0}$) regions (subdomains), with boundaries

[1] The proteins responsible for polarisation don't accumulate on the budding scars.



(a) *Schnakenberg model* **(b)** *Gierer-Meinhardt model* **(c)** *Domain with holes*

Figure 1.1: *Pattern on the flat surface. (a) Pattern with one pole for the Schnakenberg model. (b) Pattern with three poles for the Gierer-Meinhardt model. (c) Domain with holes. For parameters for (a-b) see appendix B.*

Γ_i described by a zero flux Neumann-condition ($\mathbf{n} \cdot \nabla \mathbf{c} = 0$ on Γ_i).

Besides being formed on domains containing holes, some biological patterns possess a pole (region of high concentration) at a specific region. One example of this is the polarisation of the protein Cdc42 in *S. cerevisiae*, where polarisation sites (poles) are created close to already existing budding scars (holes) [9]. Previous studies indicate that confinement of a pole to a specific region in a RD-model could be achieved by varying the model parameter values in a sub-region [5, 10, 11]. Besides varying the parameters, disturbing the initial conditions of the RD-system by a large positive noise in a sub-region could potentially confine a pole. As the effect of holes on RD-models is unclear though, it is hard to predict if any of these approaches could work. Still, as it is possible to simulate the system in Eq. (1.1) on a domain with holes, these approaches for controlling pole formation can be investigated via simulations. Consequently, with the aim to increase the understanding of RD-models on domains with holes, and how poles might be spatially confined on such a domain, this thesis investigates the following two scenarios:

1. The effect of holes on pattern formation for the Schnakenberg model and Gierer-Meinhardt model in two dimensions.
2. The effect of varying parameters and/or the initial conditions in a small sub-region when a two dimensional domain contains holes for the Schnakenberg model and Gierer-Meinhardt model in two dimensions.

2

Theory

This chapter presents a theoretical background of reaction-diffusion models, with the focus being placed on the Schnakenberg model (2.8) and Gierer-Meinhardt model (2.6). As both these models are two-species models, attention is thus placed on two-species RD-systems.

In this report vectors and matrices are denoted by small and capital letters respectively in bold font. The gradient is denoted by ∇, the laplacian operator by ∇^2, the normal vector by $\mathbf{n}$ and the scalar (inner) product by a dot $(\cdot)$.

2.1 Nondimensionalisation of RD-systems

The process of nondimensionalisation aims to remove the units in a model [4]. This is achieved by scaling the states, space-coordinates and time by a combination of parameters in the model and/or a characteristic-length. At a first glance this rescaling of the original equations might seem worthless, however this operation has two benefits. Firstly, the process of nondimensionalisation can reduce the number of model parameters [4]. This is of interest in RD-models because the pattern forming ability of a RD-model is governed by the parameter values (section 2.2). Consequently, if the number of parameters is reduced it becomes easier to find the Turing space (space of parameter values that produce a pattern). Secondly, nondimensionalisation can result in parameters that are more intuitive to interpret [4].

The technique of nondimensionalisation is in this thesis demonstrated on the Gierer-Meinhardt model. Here, the standard form of the model [8] and the nondimensionalisation procedure as in [4] are followed. The two species Gierer-Meinhardt models is:

$$\frac{\partial \alpha}{\partial t} = D_\alpha \nabla^2 \alpha + k_1 - k_2 \alpha + k_3 \frac{\alpha^2}{\beta} \tag{2.1a}$$

$$\frac{\partial \beta}{\partial t} = D_\beta \nabla^2 \beta + k_4 \alpha^2 - k_5 \beta \tag{2.1b}$$

Where D_i is diffusion constants and k_i correspond to a rate constant. The first step in the nondimensionalisation is to rescale the time and states:

$$\begin{aligned} \alpha &= \frac{k_3}{k_4}u, \quad \beta = \frac{k_3^2}{k_4 k_5}v, \quad \tau = \frac{D_\alpha t}{L^2}, \quad \mathbf{r}^* = \frac{\mathbf{r}}{L} \\ d &= \frac{D_\beta}{D_\alpha}, \quad a = \frac{k_1}{k_5}, \quad k_2 = \frac{k_2}{k_5}, \quad \gamma = \frac{L^2 k_5}{D_\alpha} \end{aligned} \tag{2.2}$$

Where u and v are the dimensionless concentrations, τ is the dimensionless time and $\mathbf{r}^*$ is the dimensionless spatial coordinate. Using the expression of τ in Eq. (2.2) it follows from the chain rule that:

$$\frac{\partial \alpha}{\partial t} = \frac{\partial \alpha}{\partial \tau}\frac{\partial \tau}{\partial t} = \frac{\partial \alpha}{\partial \tau}\frac{D_\alpha}{L^2}, \quad \frac{\partial \beta}{\partial t} = \frac{\partial \beta}{\partial \tau}\frac{\partial \tau}{\partial t} = \frac{\partial \beta}{\partial \tau}\frac{D_\alpha}{L^2} \tag{2.3}$$

Furthermore, by applying the chain rule, product rule and lastly chain rule it follows that for component r_i in $\mathbf{r}$ that:

$$\frac{\partial^2 \alpha}{\partial r_i^2} = \frac{\partial^2 \alpha}{\partial (r_i^*)^2}\left(\frac{\partial r_i^*}{\partial r_i}\right)^2 + \frac{\partial r}{\partial r_i^*}\underbrace{\left(\frac{\partial^2 r_i^*}{\partial r_i^2}\right)}_{=0} = \frac{\partial^2 r}{\partial (r_i^*)^2}\frac{1}{L^2} \tag{2.4}$$

Now using Eq. (2.3) and Eq. (2.4) it follows that the system in Eq. (2.1) can be rewritten as:

$$\begin{aligned} \frac{\partial \alpha}{\partial \tau} &= \nabla_*^2 \alpha + \frac{L^2}{D_\alpha}\left(k_1 - k_2\alpha + K_3\frac{\alpha^2}{\beta}\right) \\ \frac{\partial \beta}{\partial \tau} &= \underbrace{\frac{D_\beta}{D_\alpha}}_{=d}\nabla_*^2 \beta + \frac{L^2}{D_\beta}\left(k_4\alpha^2 - k_5\beta\right) \end{aligned} \tag{2.5}$$

By now substituting α for u and β for v using the substitutions in Eq. (2.2), and dropping the stars on ∇_*^2 for ease of notation, it follows that the system in Eq. (2.1) can be rewritten as:

$$\frac{\partial u}{\partial \tau} = \nabla^2 u + \gamma\left(a - bu + \frac{u^2}{v}\right) \tag{2.6a}$$

$$\frac{\partial v}{\partial \tau} = d\nabla^2 v + \gamma(u^2 - v) \tag{2.6b}$$

For the Schnakenberg model [7], the general form is:

$$\frac{\partial \alpha}{\partial t} = D_\alpha \nabla^2 \alpha + k_1 - k_2\alpha + k_3\alpha^2\beta \tag{2.7a}$$

$$\frac{\partial \beta}{\partial t} = D_\beta \nabla^2 \beta + k_4\beta^2 - k_5\alpha^2\beta, \tag{2.7b}$$

In a similar manner, it can be shown (for details see [4]) that the corresponding nondimensional model is:

$$\frac{\partial u}{\partial \tau} = \nabla^2 u + \gamma(a - u + u^2 v) \tag{2.8a}$$

$$\frac{\partial v}{\partial \tau} = d\nabla^2 v + \gamma(b - u^2 v) \tag{2.8b}$$

By comparing the original and dimensionless model formats three noteworthy details can be noted. Firstly, for both models the nondimensionalisation reduced the number of parameters from seven to four, which simplifies the analysis performed in section 2.2 and 2.2.3. Secondly, for both models the diffusion constant (D_α and D_y) were replaced by d, a parameter easier to interpret than the individual diffusion constants. Thirdly, it can be noted that both models after nondimensionalisation have the same general format:

$$\frac{\partial u}{\partial \tau} = \nabla^2 u + \gamma f(u, v) \tag{2.9a}$$

$$\frac{\partial v}{\partial \tau} = d\nabla^2 v + \gamma g(u, v) \tag{2.9b}$$

Where γ has the interpretation of a control parameters that determines the strength of the reaction terms relative to the diffusion d [4].

2.2 Turing patterns

The ability of RD-model, like the one in Eq. (1.1), to generate patterns strongly depends on the reaction dynamics $\mathbf{f}(\mathbf{c})$ (with associated parameters) and the diffusion constants. It has been shown that so called Turing patterns can be formed if a model fulfils diffusion driven instability; the homogeneous steady state is stable to small perturbations in the absence of diffusion, but unstable in presence of diffusion [4]. The ability of a model to fulfil diffusion driven instability strongly depends on the model parameters, and the region where the parameters fulfil this is referred to as the Turing space. As this thesis considers two species models, this section derives the mathematical conditions for diffusion driven instability for a two species model (Eq. (2.25)). In addition, based on these conditions an intuitive explanation to pattern formation is provided. Lastly, based on the conditions the actual Turing spaces for the Schnakenberg model and the Gierer-Meinhardt model are presented (Fig. 2.2.3).

2.2.1 Mathematical conditions for Turing patterns

The derivation of the conditions for diffusion driven instability presented in this section will relatively closely follow the derivation presented by Murray [4]. Furthermore, since this thesis investigates two species models the conditions are derived for the following general two-species RD-system:

$$\begin{aligned} \frac{\partial u}{\partial t} &= D_u \nabla^2 u + F(u,v), \quad \mathbf{n} \cdot \nabla u = 0 \quad \text{on } \Gamma \\ \frac{\partial v}{\partial t} &= D_v \nabla^2 v + G(u,v), \quad \mathbf{n} \cdot \nabla v = 0 \quad \text{on } \Gamma \end{aligned} \tag{2.10}$$

Where Γ refers to the boundary. Note that in this general format a factor γ cannot be factorised out from the reaction terms. The first requirement for diffusion driven instability is that the steady state (u^*, v^*) is stable given no diffusion. With other words, a small perturbation to the steady state $\delta\mathbf{w} = (u-u^*, v-v^*)$ should over time converge to the steady state. Conditions for when this holds can be investigated using linear stability analysis, since a linearisation approximates the diffusion free system well for small perturbations:

$$\frac{\mathrm{d}\delta\mathbf{w}}{\mathrm{d}t} \approx \mathbf{J}^* \delta\mathbf{w} \tag{2.11}$$

Where $\mathbf{J}^*$ is the Jacobian matrix evaluated at the steady state:

$$\mathbf{J}^* = \begin{bmatrix} \frac{\partial F}{\partial u} & \frac{\partial F}{\partial v} \\ \frac{\partial G}{\partial u} & \frac{\partial G}{\partial v} \end{bmatrix} \Bigg|_{(u^*,v^*)} = \begin{bmatrix} F_u^* & F_v^* \\ G_u^* & G_v^* \end{bmatrix} \tag{2.12}$$

As shown in Appendix A, the system in Eq. (2.11) has the following general solution:

$$\delta\mathbf{w}_t = c_1 \mathbf{a}_1 e^{\lambda_1 t} + c_2 + c_2 \mathbf{a}_2 e^{\lambda_2 t} \tag{2.13}$$

Where λ_1 and λ_2 are the eigenvalues of $\mathbf{J}^*$. The eigenvalues can be obtained by solving the characteristic equation $\det(\mathbf{J}^* - \mathbf{I}\lambda) = 0$, resulting in:

$$\lambda_1, \lambda_2 = \frac{1}{2}\left((F_u^* + G_v^*) \pm \left((F_u^* + G_v^*)^2 - 4(F_u^* G_v^* - F_v^* G_u^*) \right)^{1/2} \right) \tag{2.14}$$

From Eq. (2.13) it follows that $\lambda_1, \lambda_2 < 0$ in order for the steady state to be stable, since this will result in small perturbations $\delta w \to 0$ as $t \to \infty$ (the initial perturbation converges to the steady state). By investigating Eq. (2.14) it is noted that the following two conditions must hold for this to be true:

$$\mathrm{Tr}(\mathbf{J}^*) = F_u^* + G_v^* < 0, \quad \det(\mathbf{J}^*) = F_u^* G_v^* - F_v^* G_u^* > 0 \tag{2.15}$$

The next condition for Turing patters is diffusion driven instability. More specifically, a small perturbation $\delta\mathbf{w}$ to the steady state should result in the dynamics moving away from the steady state when diffusion is present. Adding diffusion to the linearised dynamics results in the following system:

$$\frac{\partial \delta\mathbf{w}}{\partial t} = \mathbf{J}^* \delta\mathbf{w} + \mathbf{D}\nabla^2 \delta\mathbf{w} \tag{2.16}$$

Where $\mathbf{D} = \text{diag}(D_u, D_v)$ is the diagonal diffusion matrix. The solutions to the system in Eq. (2.16), following the method of variable separation, is [4]:

$$\delta\mathbf{w} = \sum_k \mathbf{c}_k e^{\alpha_k t}\mathbf{W}_k(\mathbf{r}) \tag{2.17}$$

Where $\mathbf{c}_k$ are given by the Fourier series expansion of the initial conditions and $\mathbf{W}_k(\mathbf{r})$ is an eigenfunction solving the spatial problem $\nabla\mathbf{W}_k + k^2\mathbf{W}_k = 0$ that arises from the variable separation [4]. From Eq. (2.17) it follows that in order to have diffusion driven instability there must exist at least one $\alpha_k > 0$ (else the dynamics will not move away from the steady state). To obtain an expression of α_k Eq. (2.17) can be inserted into Eq. (2.16), resulting in the following equation [4]:

$$\alpha_k\mathbf{W}_k = \mathbf{J}^*\mathbf{W}_k + \underbrace{\mathbf{D}\nabla^2\mathbf{W}_k}_{=-k^2\mathbf{D}\mathbf{W}_k} \implies (\mathbf{J}^* - k^2\mathbf{D})\mathbf{W}_k = \alpha_k\mathbf{W}_k \tag{2.18}$$

From Eq. (2.18) it follows that the α_k value can be obtained by solving for the eigenvalues of the matrix

$$\mathbf{M} = \mathbf{J}^* - k^2\mathbf{D} = \begin{bmatrix} F_u^* - k^2 D_u & F_v^* \\ G_u^* & G_v^* - k^2 D_v^2 \end{bmatrix}. \tag{2.19}$$

By applying some algebra on the characteristic equation $\det(\mathbf{M} - \alpha\mathbf{I}) = 0$ the eigenvalues can, for a fixed k, be formulated as:

$$\alpha_1, \alpha_2 = \frac{\text{Tr}(\mathbf{M}) \pm \sqrt{\text{Tr}(\mathbf{M})^2 - 4\det(\mathbf{M})}}{2} \tag{2.20}$$

As $\text{Tr}(\mathbf{M}) < 0$, due $\text{Tr}(\mathbf{J}^*) < 0$ and $k, D_u, D_v > 0$, it follows that the only way to achieve diffusion driven instability is via $\det(\mathbf{M}) < 0$. A bit of algebra yields that:

$$\det(\mathbf{M}) = \det(\mathbf{J}^*) - k^2(D_v F_u^* + D_u G_v^*) + k^4 D_u D_v \tag{2.21}$$

Overall it follows that $\det(\mathbf{M})$ is a parabolic function with respect to k^2. Some straightforward derivation yields a minimum for this function at:

$$k_{\text{min}}^2 = \frac{D_v F_u^* + D_u G_v^*}{2D_u D_v} \tag{2.22}$$

Inserting this minimum in Eq. (2.21) yields:

$$\det(\mathbf{M})_{\text{min}} = \det(\mathbf{J}^*) - \frac{(D_v F_u^* + D_u G_v^*)^2}{4D_u D_v} \tag{2.23}$$

As $\det(\mathbf{J}^*) > 0$ and $k_{\text{min}} > 0$, it follows that the following two conditions must be fulfilled for $\det(\mathbf{M})_{\text{min}} < 0$ to be true:

$$0 < D_v F_u^* + D_u G_v^*, \quad \det(\mathbf{J}^*) < \frac{(D_v F_u^* + D_u G_v^*)^2}{4 D_u D_v} \tag{2.24}$$

If the conditions in Eq. (2.24) are fulfilled it follows that there exists a k such that $\det(\mathbf{M})_{\min} < 0$ and, in extension via Eq. (2.21), there exists an $\alpha_k > 0$. In other words, the system exhibits diffusion driven instability. To simplify further calculations, it can be noted that for models of the format in Eq. (2.9), the conditions in Eq. (2.15) and Eq. (2.24) simplify into:

$$f_u^* + g_v^* < 0 \tag{2.25a}$$
$$f_u^* g_v^* - f_v^* g_u^* > 0 \tag{2.25b}$$
$$d f_u^* + g_v^* > 0 \tag{2.25c}$$
$$f_u^* g_v^* - f_v^* g_u^* < \frac{(d f_u^* + g_v^*)^2}{4d} \tag{2.25d}$$

For the models investigated in this thesis the conditions in Eq. (2.25) give four necessary conditions for diffusion driven instability. In extension, they therefore make up necessary conditions for Turing pattern formation. If the parameters in a model fulfil these conditions, it means that given a random perturbation of the steady state (u^*, v^*) the eigenfunctions connected with the eigenvalues $\alpha_k > 0$ will grow. As the reaction term $\mathbf{f}(\mathbf{c})$ generally is bounded this growth will not be indefinite, instead the growing eigenfunctions might settle into spatial patterns [4].

2.2.2 An intuitive description of Turing pattern formation

It is possible from the conditions in Eq. (2.25) to get an intuitive feeling of Turing pattern formation. By assuming that $f_u^* > 0$, some straightforward algebra on Eq. (2.25) result in $d > 1$ and two possible sign configurations of $\mathbf{J}^*$ (Fig. 2.1). The two cases in Fig. 2.1 have two traits in common; u is self-activating around the steady state ($f_u^* > 0$) and v diffuses faster than u ($d > 1$) [4]. Besides these similarities the two configurations are different, and they actually correspond to distinct mechanisms of pattern formation [4]. For the Gierer-Meinhardt model (Fig. 2.1a) and the Schnakenberg model (Fig. 2.1b) the following two paragraphs present an intuitive description, based on those provided by Murray [4], for each model.

$$\begin{bmatrix} + & - \\ + & - \end{bmatrix} \qquad \begin{bmatrix} + & + \\ - & - \end{bmatrix}$$

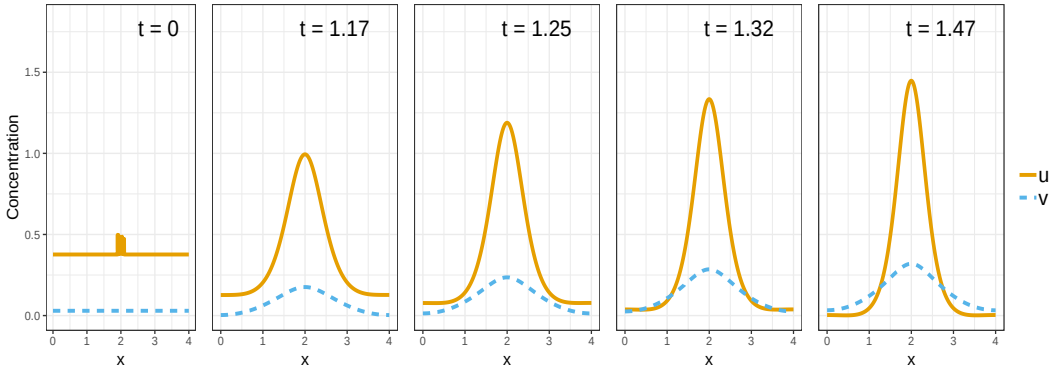
(a) *Gierer-Meinhardt model* **(b)** *Schnakenberg model*

Figure 2.1: *The two configurations of* $\mathbf{J}^*$ *that gives rise to Turing patterns for a two-species RD-model. (a) Gierer-Meinhardt model. (b) Schnakenberg model.*

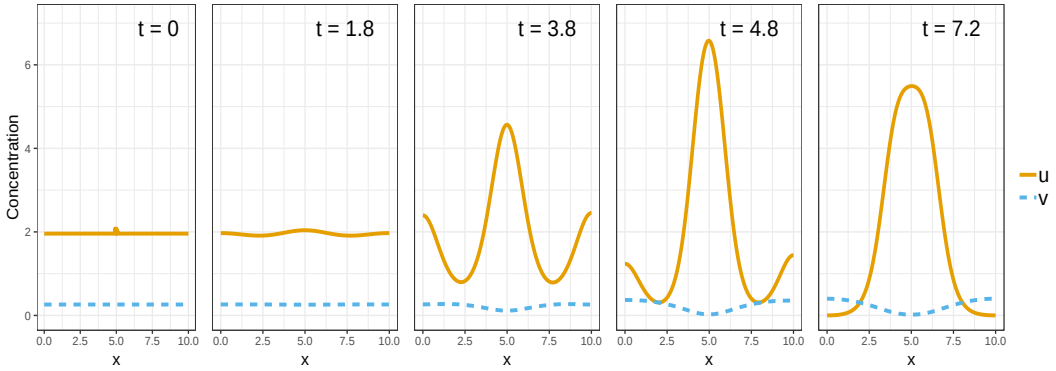
Considering the Gierer-Meinhardt case (Fig. 2.1a), it corresponds to a scenario where around the steady state, the self activating component u activates v ($g_u^* > 0$).

Consequently, if u is perturbed around the steady state in a small region, the levels of u and v will both increase in that region. As $d > 1$, v diffuses faster away from this zone of local activation than u, and as v inhibits u ($f_v^* < 0$) this results in the creation of a barrier that inhibits formation of u outside of the activation zone. Overall, this results in that if u is perturbed randomly around the steady state, regions can arise with high levels of u and v, interspersed with regions were both concentrations are low. This entire process is illustrated in Fig. 2.2a for the one dimensional case.

Considering the Schnakenberg case (Fig. 2.1b), it corresponds to a scenario where around the steady state the self activating component u inhibits the formation of v ($g_u^* < 0$). Consequently, if u is perturbed around the steady state in a small region, the levels of u increase, while the levels of v decrease in that region. As $d > 1$, this decrease in v results in a net flux of v from the neighbouring regions, leading to a decrease of u in the those regions as the activation of u depends on v ($f_v^* > 0$). Hence, if u is perturbed randomly around the steady state this mechanism can give rise to regions with high u and low v levels, intermixed with regions of low u and high v levels. This entire process is illustrated in Fig. 2.2b for the one dimensional case.



(a) $\mathbf{J}^*$ *sign proportional to case Fig. 2.1a (Gierer-Meinhardt model)*



(b) $\mathbf{J}^*$ *sign proportional to Fig. 2.1b (Schnakenberg model)*

Figure 2.2: *1D-illustration of how Turing patterns are formed for different sign-configuration of* $\mathbf{J}^*$*. In both plots the homogeneous steady state of* u *is randomly perturbed in the middle of the domain. The parameter values used for the Gierer-Meinhardt model (a) and Schnakenberg model (b) can be found in appendix B*

2.2.3 Finding the Turing space

The conditions in Eq. (2.25) define the Turing space, that is the set of all parameters that can give rise to Turing patterns [4]. As Turing patterns cannot be generated without knowing a subset of the Turing space, it is relevant to have a methodology for identifying this space. One possible way to identify it is to make a numeric grid-search over a set of possible parameter values, and then report those parameter combinations which fulfil the conditions in Eq. (2.25). This approach is straightforward, but as the Turing space in this case depend on a, b and d it becomes tricky to visualise the result from such a grid-search. In the case of the Gierer-Meinhardt model and the Schnakenberg model another possible approach is to analytically derive the Turing space in a manner that can be visualised [4]. Although this method algebraically heavy, it is the preferred choice in this thesis as it avoids the mentioned visualisation problem.

The technique of deriving the Turing space was first introduced by Murray [4], and is demonstrated in this thesis on the Gierer-Meinhardt model. The first step in analytically finding the Turing space is to express the steady state $f(u, v) = g(u, v) = 0$ in terms of u^* and the parameters a and b:

$$\begin{aligned} a - bu + \frac{u^2}{v} = 0 \implies u^* &= \frac{a+1}{b} \\ u^2 - v = 0 \implies v^* &= \frac{(a+1)^2}{b^2} = (u^*)^2 \end{aligned} \tag{2.26}$$

As noted in Eq. (2.26) the Gierer-Meinhardt only has one steady state. If the model had more than one, the following steps would have to be repeated for each one. The next step in finding the Turing space is to express the different conditions in Eq. (2.25) in terms of a and u^*. To simplify further calculations it is first advantageous to rewrite the Jacobian as:

$$\mathbf{J}^* = \begin{bmatrix} f_u^* & f_v^* \\ g_u^* & g_v^* \end{bmatrix} = \begin{bmatrix} -b + 2\frac{u^*}{v^*} & -\frac{(u^*)^2}{(v^*)^2} \\ 2u^* & -1 \end{bmatrix} = \begin{bmatrix} \frac{1-a}{u^*} & -\frac{1}{(u^*)^2} \\ 2u^* & -1 \end{bmatrix} \tag{2.27}$$

Where b was rewritten using the steady state in Eq. (2.26). Starting with the first condition in Eq. (2.25a), it can be rewritten as:

$$f_u^* + g_v^* = \frac{1-a}{u^*} - 1 < 0 \implies a > 1 - u^* \tag{2.28}$$

The second conditions in Eq. (2.25b) holds trivially:

$$|\mathbf{J}^*| = f_u^* g_v^* - f_v^* g_u^* = \frac{a-1}{u^*} + \frac{2}{u^*} = \frac{1+a}{u^*} > 0 \tag{2.29}$$

The third condition in Eq. (2.25c) yields:

$$df_u^* + g_v^* = d\frac{1-a}{u^*} - 1 > 0 \implies a < 1 - \frac{u^*}{d} \tag{2.30}$$

The last condition in Eq. (2.25d) yields:

$$|\mathbf{J}^*| < \frac{(df_u + g_v)^2}{4d} \implies \frac{1+a}{u^*} < \frac{\Big(d(1-a) - u^*\Big)^2}{4d(u^*)^2} \tag{2.31}$$

Now by applying some algebra on Eq. (2.31) it follows that:

$$\begin{aligned} 0 &< \frac{a^2d^2 - 2ad^2 - 2adu + d^2 - 6du + u^2}{4du^2} \underset{d>0}{\implies} \\ 0 &< a^2 \underbrace{d^2}_{=c_1(d)} + a\underbrace{(-2d^2 - 2du)}_{=c_2(d)} + \underbrace{d^2 + u^2 - 6du}_{=c_3(d)} \end{aligned} \tag{2.32}$$

Overall this results in the following constraints on a:

$$a < \frac{-c_2 - \sqrt{c_2^2 - 4c_1c_3}}{2c_1}, \quad a > \frac{-c_2 + \sqrt{c_2^2 - 4c_1c_3}}{2c_1} \tag{2.33}$$

Which after some algebra simplifies to:

$$a < 1 + \frac{du^* - 2\sqrt{8d^3u^*}}{d^2}, \quad a > 1 + \frac{du^* + 2\sqrt{8d^3u^*}}{d^2} \tag{2.34}$$

The rightmust condition in Eq. (2.34) cannot hold as it is directly conflicting with Eq. (2.31), resulting in the first condition in Eq. (2.34) being required to hold. Overall, Eq. (2.28), (2.30) and (2.34) result in the following limits of the Turing space for the Gierer-Meinhardt model:

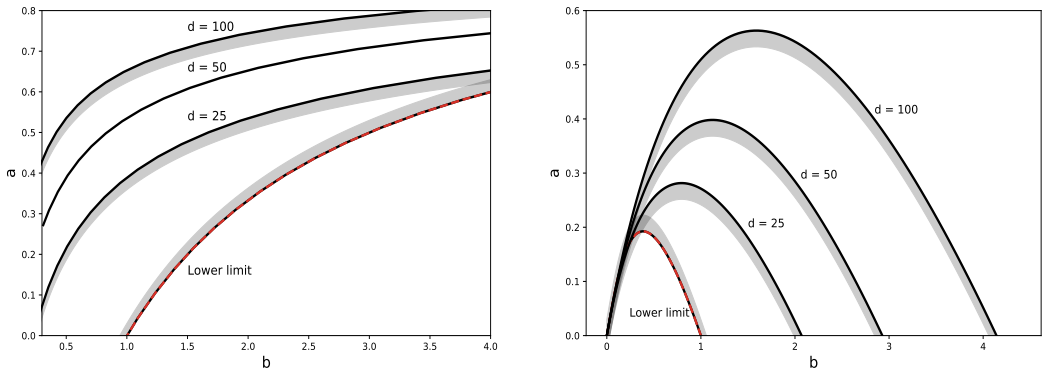
$$a_{\min} = 1 - u^* \tag{2.35a}$$

$$a_{\max} = \min\left\{1 - \frac{u^*}{d}, 1 + \frac{du^* - \sqrt{8d^3u^*}}{d^2}\right\} \tag{2.35b}$$

For the Schnakenberg model (for details see [4]), the following limits hold for the Turing space:

$$a_{\min} = \frac{u^*}{2}\Big(1 - (u^*)^2\Big) \tag{2.36a}$$

$$a_{\max} = \frac{u^*}{2}\Big(1 - \frac{2u^*}{\sqrt{d}} - \frac{(u^*)^2}{d}\Big) \tag{2.36b}$$



(a) *Gierer-Meinhardt model* **(b)** *Schnakenberg model*

Figure 2.3: *The Turing space for the Gierer Meinhardt model (a) and the Schnakenberg model (b) for different values of d. For example, if $d = 25$ in (a) any combination of (a, b) being below the curve marked by $d = 25$ and above the lower limit (red-dotted curve) will result in diffusion driven instability.*

Given the limits in Eq. (2.35) and (2.36) it is possible to plot the Turing space. This is done by noting that $b = (a + 1)/u^*$ for the Gierer-Meinhardt model and that $b = u^* - a$ for the Schnakenberg model. Given this, the parameter space can be plotted for different d-values by plotting the resulting $(a_{\max/\min}, b_{\max/\min})$ values for each u^* in the interval $[0, \infty)$. This procedure results in curves that define the Turing space (Fig. 2.3). A noteworthy detail about these curves however is that they do not guarantee Turing patterns. This is because the formation of patterns also depend on γ, as it has to be big enough to allow any eigenfunction to grow [4].

3

Methods

This chapter presents the finite element variational formulation (Eq. (3.5)) used for solving the Gierer-Meinhardt model and the Schnakenberg model on a domain containing holes. In addition the finite element meshes, on which the PDEs were solved, are presented.

All the code used for solving the PDE:s, generating the meshes and generating the figures for this thesis can be found on GitHub (`https://github.com/sebapersson/Master_thesis`). Some important details about the code is that must figures were created using R (version 3.6.2), the meshes were generated using Gmsh (version 4.4.1) [12], and the PDE:s were solved in Python (version 3.7.5) using FeniCS [13]. Efforts have been made to make the project as reproducible as possible by basing the directory structure on the following two suggestions [14, 15]. Assuming a unix based operating system and an Anaconda distribution, the results should be reproducible by first creating an Anaconda environment form the yml-file, and then given this environment run the *Run_all.sh*-script. It is recommended to run the *Run_all.sh*-script on a cluster, as it took over 60 hours to produce the result in this report on a cluster with 112 CPU:s (Intel(R) Xeon(R) Platinum 8180 CPU @ 2.50GHz). More details about reproducing the results can be found on the GitHub page.

3.1 Finite element variational formulation

The finite element method (FEM) is a commonly used approach for solving PDE:s. It is the preferred method in this thesis by two reasons. Firstly, it is relatively easy to adapt the method to complex geometries (e.g domains with holes) [16]. Secondly, it is relatively straightforward to implement a FEM-solver using the open-source software FeniCS, which only requires the finite element variational formulation as input [13].

The underlying idea of FEM is to rewrite the PDE-system into a variational (or weak) formulation. Following this, a FEM-solver aims to approximate a solution to the variational formulation by solving the approximate problem which is the finite element variational formulation [13, 16]. In order to avoid repeating the same calculations twice, this sections provides the finite element variational formulation for the general system in Eq. (2.9). Before introducing the variational formulation however, some notations must be introduced. Let Ω correspond the the domain where the species u and v can diffuse, and let Γ denote the outer boundary of Ω.

Let ω_i correspond to hole i and let Γ_i be the boundary between ω_i and Ω (Fig 1.1c). Assuming the holes to be zero-concentration regions $(u, v) = \mathbf{0}$ bounded by a zero flux Neumann condition, the PDE-system of interest to solve in this thesis can be formulated as:

$$\frac{\partial u}{\partial \tau} = u_\tau = \nabla^2 u + \gamma f(u, v), \qquad u \in \Omega \tag{3.1a}$$

$$\frac{\partial v}{\partial \tau} = v_\tau = d\nabla^2 u + \gamma g(u, v), \qquad u \in \Omega \tag{3.1b}$$

$$u = v = 0, \qquad u, v \in \omega_i, \quad \forall i \tag{3.1c}$$

$$\mathbf{n} \cdot \nabla u = \mathbf{n} \cdot \nabla v = 0, \qquad u, v \in \Gamma \cup \Gamma_i \quad \forall i \tag{3.1d}$$

Given the system in Eq. (3.1) it is possible to formulate the finite element variational formulation. The variational formulation can be obtained for a fixed τ by multiplying each equation by a test functions $q, s \in \mathcal{H}^1(\Omega)$ (Sobolev space see [16] for definition), integrating over Ω and adding Eq. (3.1a) and Eq. (3.1b) together. By this procedure the variational formulation becomes; for a fixed τ find $u, v \in \mathcal{H}^1(\Omega)$ such that:

$$\begin{aligned} &\int_\Omega \Big(u_\tau q - \gamma f(u, v) q + \nabla u \cdot \nabla q \Big) \mathrm{d}\mathbf{x} + \\ &\int_\Omega \Big(v_\tau s - \gamma g(u, v) s + d\nabla v \cdot \nabla s \Big) \mathrm{d}\mathbf{x} = 0 \quad \forall s, q \in \mathcal{H}^1(\Omega) \end{aligned} \tag{3.2}$$

As mentioned, the idea behind FEM is to find an approximate solution (u_h, v_h) to Eq. (3.2). There are many different ways to approximate the solution [13], and as the must simple approach works for the system in Eq. (3.1) it is the one used in this thesis. This approach corresponds to first partitioning the domain into triangles. More precisely, let $\mathcal{T}_h = \{K\}$ be the set of triangles such that an intersection is either an edge, corner or empty. Given this, the approximate solution to (3.2) is obtained by replacing $\mathcal{H}^1(\Omega)$ in Eq. (3.2) by:

$$\mathcal{S}_h = \{v \in \mathcal{C}(\bar{\Omega}), v|_k \in \mathcal{P}_1(k), \ \forall K \in \mathcal{T}_h\} \tag{3.3}$$

Where $\mathcal{C}$ is the set of continuous function in this case on $\bar{\Omega}$ (union of Ω and the boundaries) and $\mathcal{P}_1(K)$ is the set of linear functions on K:

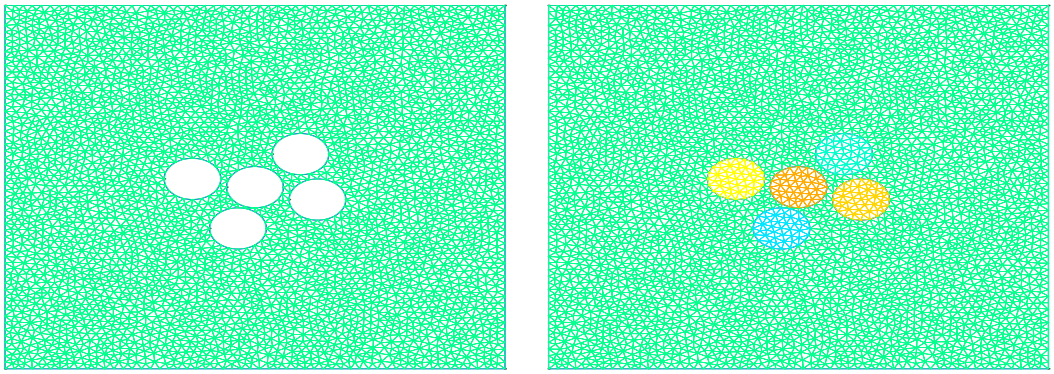
$$\mathcal{P}_1(K) = \{v : v = c_0 + c_1 x_1 + c_2 x_2, (x_1, x_2) \in K, c_0, c_1, c_2 \in \mathbb{R}\} \tag{3.4}$$

By seeking a solution in $\mathcal{S}_h$ to (3.2) it is possible, for a fixed τ, to obtain an approximate piecewise linear spatial solution to Eq. (3.1). However, as noted in Eq. (3.1) the problem is also time-dependent. On other words, in order to solve the problem a time-discretisation is required. In this thesis an explicit-implicit time-discretisation method is used, where the reaction terms $f(u, v)$ and $g(u, v)$ are made explicit while the gradient is kept implicit. Overall, given this time-discretisation the finite element variational formulation becomes; for each τ find $u_h^{(n)}, v_h^{(n)} \in \mathcal{S}_h(\Omega)$, where the superscript denote time-step n, such that:

$$\int_\Omega \Big(\Delta\tau^{-1}(u_h^{(n+1)} - u_h^{(n)})q - \gamma f(u_h^{(n)}, v_h^{(n)})q + \nabla u_h^{(n+1)} \cdot \nabla q \Big) \mathrm{d}\mathbf{x} +$$
$$\int_\Omega \Big(\Delta\tau^{-1}(v_h^{(n+1)} - v_h^{(n)})s - \gamma g(u_h^{(n)}, v_h^{(n)})s + \nabla v_h^{(n+1)} \cdot \nabla s \Big) \mathrm{d}\mathbf{x} = 0 \quad \forall s, q \in \mathcal{S}_h(\Omega) \tag{3.5}$$

Given an initial condition $(u_h^{(0)}, v_h^{(0)})$ the system in Eq. (3.5) can be plugged into FeniCS and solved for each time step.

Lastly, it should be noted that there are two ways to implement the holes in Eq. (3.1) for the variational finite element formulation in Eq. (3.5). The first approach is to simply make holes in the mesh (Fig 3.1a) and the second approach it to consider holes as sub-domains (Fig 3.1b). Although the second approach is much more engaging from a coding-perspective (the zero concentration condition for the sub-domain must be added to the code), it is the preferred choice in this thesis as it lays a more general foundation for future studies. This is because if the holes are sub-domains, it becomes relatively straightforward to investigate other scenarios, like what happens if the species are allowed to diffuse, but not partake in any reaction in a hole.



(a) *Holes as holes in the mesh* **(b)** *Holes as sub-domains*

Figure 3.1: *Two possible ways to implement a domain with holes. (a) Implement the holes as holes the mesh. (b) Consider the holes as sub-domains (the regions with different colour).*

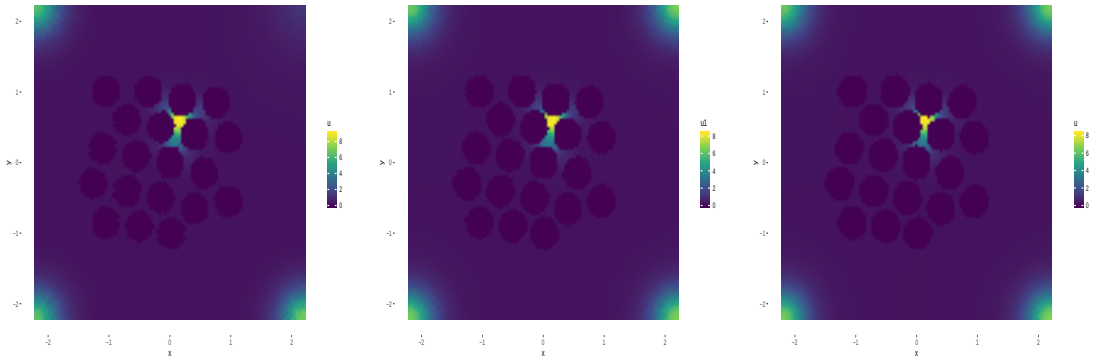
3.2 Finite element meshes

As noted in section 3.1 a partition of the domain (mesh), in this case consisting of triangles, is required for solving the system in Eq. (3.5). In this thesis the meshes were created using Gmsh [12]. The main motivation for using Gmsh, instead of the easier to use FeniCS mesh-generator, is that the holes were modelled as sub-domains. With Gmsh it is possible to take these sub-domains into account and create a mesh where the nodes create a smooth line between the different domains (Fig. 3.1b), which is not possible with FeniCS. Furthermore, it is possible to import

into to FeniCS from Gmsh information of which sub-domain a node belongs to. This removes the need of defining the sub-domains via geometric conditions in the code, which would have been required if the mesh was generated in FeniCS.

Eight different meshes were created (Fig. C.1). Out of these, four had a circular (radius = 2.5) base-domain and the other four had a rectangular base domain (size 4.2×4.2). For each base-domain a mesh with zero, five, seven and 20 holes was created. The holes were circular and densely packed, and the area of each hole was 1.5 % of of the base-domain area. These hole characteristics were chosen since small densely packed holes have been observed in biological systems [17].

For the eight different meshes the characteristic length (lc), which determines the coarseness of the mesh [12], was fine-tuned. This parameter is of importance to tune properly, as a small lc-value results in a small numerical error but long runtime, while a large lc-value results in a large error but short runtime [16]. Thus to avoid numerical errors, but ensuring a reasonable runtime, a good lc-value is required. The lc-value was tuned by a procedure where the steady state was perturbed in the centre of the domain by a constant value. The idea behind this procedure is that if the mesh is fine enough, then making it finer should not affect the result. Overall three different lc-values were tried, $lc = (0.06, 0.04, 0.03)$. Generally $lc = 0.06$ worked well, but obvious errors were produced for the Gierer-Meinhardt model when using a mesh with 20 holes (Fig. 3.2). Regarding a lc-value of 0.04, it yielded the same result for both models and all meshes as a lc-value of 0.03 (only the case in Fig. 3.2 shown here). Consequently, all the meshes were constructed with $lc = 0.04$.



(a) $lc = 0.06$ **(b)** $lc = 0.04$ **(c)** $lc = 0.03$

Figure 3.2: *The Gierer-Meinhardt model at $\tau = 1.5$ for identical domains with different lc-values (smaller value corresponds to a finer mesh). In each case the initial steady state was perturbed by a constant value in the middle of the domain. As noted, $lc = 0.06$ deviates in the top-right corner from the other cases. The parameter values used can be found in appendix B*

4

Result

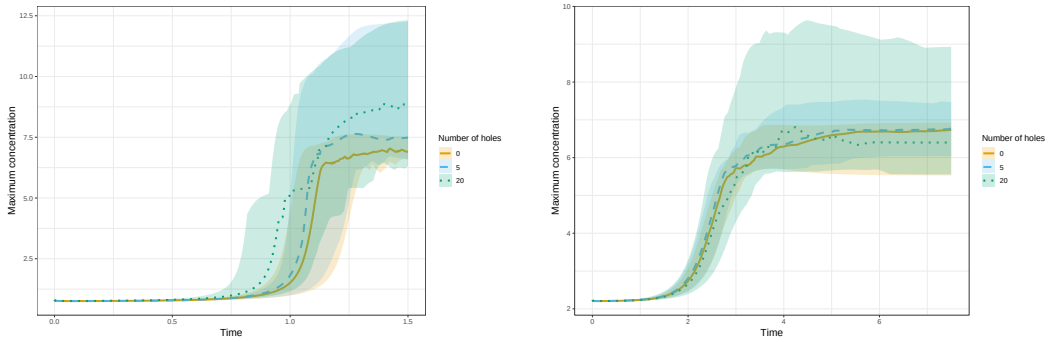
Many different experiments were conducted in order to investigate the effect of holes on pattern formation for the Gierer-Meinhardt model and the Schnakenberg model. As mentioned in chapter 1, the first point of interest to investigate in these experiments was the general effect of holes on pattern formation. This was investigated by observing the shape of the created patterns and the pattern-creation time (section 4.1). The second point of interest to investigate was the possibility of spatially confining the poles (regions of high concentrations) in the created patterns. This was investigated by observing what happened when the initial homogeneous steady state was varied in a sub-region (section 4.2) and by varying the parameters in a sub-region (section 4.3). The parameter values used for the plots mentioned in this chapter can be found in appendix B.

Two different base-domains were created for the meshes, one rectangular (size 4.2×4.2) and one circular (radius = 2.5) (section 3.2). As the results generated on the rectangular base-domain were highly similar to those for the circular base-domain, this chapter will only report results from the rectangular one. Lastly, the holes considered are closely packed circular holes whose area corresponds to 1.5 % of the base-domain's area.

4.1 Effect of holes

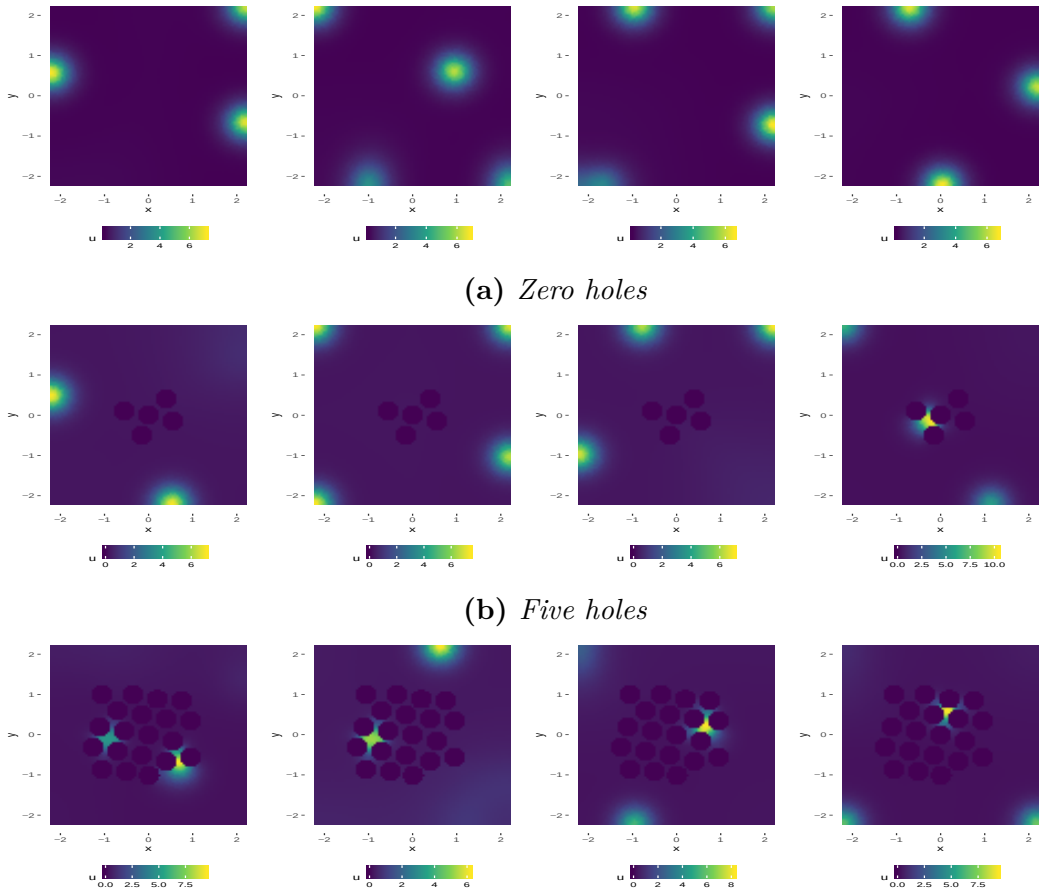
To investigate the effect of holes on pattern formation 30 simulations, with different random perturbations of the initial steady state, were each run on the the domains with zero, five and 20 holes. Based on these simulations, the shape of the created patterns and the pattern creation time were investigated. Starting with the pattern creation time, it was investigated by plotting the maximum value of u (Fig. 4.1). As patterns show regions of high concentration, the maximum value of u acts as an indicator of whether or not a pattern has been created. From observing the maximum u value (Fig. 4.1) it is noted that although there is quite a spread in the data, the Gierer-Meinhardt patterns seems to be created faster when the domain contain more holes. Furthermore, for both models it appears that poles with a higher concentration can be created when the domain contains holes.

To investigate the shape of created pattern, four out of the 30 simulations for each domain (zero, five and 20 holes), were plotted at the end time steady state for the Gierer-Meinhardt model (Fig. 4.2). Starting with comparing the control case of



(a) *Gierer-Meinhardt model* **(b)** *Schnakenberg model*

Figure 4.1: *The maximum concentration of u for zero, five and 20 holes for the Gierer-Meinhardt and the Schnakenberg model. The data is generated from ten simulations with different random perturbations of the initial steady state. The lines correspond to the median, and the bands to 5 % and 95 % quantiles.*



(a) *Zero holes*

(b) *Five holes*

(c) *20 holes*

Figure 4.2: *The u value for zero, five and 20 holes for the Gierer-Meinhardt model at $\tau = 2.0$. The data is generated from four simulations with different random perturbations of the initial steady state.*

zero holes with that of five holes, it appears that the holes have a small effect (Fig. 4.2a and 4.2b). The most noteworthy difference occurs in one of the cases with five holes (the rightmost one) as a pole is formed in the dense hole region at a higher concentration than the other poles. This indicates that on a domain with five holes, sometimes poles can be created in the dense hole region. Furthermore, it appears that if a pole is created in the dense hole region it can have a higher concentration compared to the poles created on the zero holes domain. A more distinct difference appears when comparing the zero and 20 holes cases (Fig. 4.2a and 4.2c). In the case of 20 holes, a pole is always formed close to, or directly in, the region that is dense in holes. In addition, in most cases when a pole is formed in the dense hole region it has a higher concentration than the zero holes poles. To verify that this behaviour was not due to chance, four other simulations for the 20 holes domain were inspected at $\tau = 2.0$. Overall, the same behaviour was observed (Fig. C.2). For the Schnakenberg model a pole was created in connection to dense hole region more frequently for the five holes domain. Besides this a rather similar result holds (Fig. C.3). That is, for 20 holes a pole often seems to be formed close to, or directly in, the region that is dense in holes and sometimes, although it does not appear to be as frequent as for the Gierer-Meinhardt model, such a pole has a higher concentration. Overall the results suggest that if the domain has a dense region of many holes, such as 20 holes, the self-activating component u can accumulate in that dense region. Furthermore, Fig. 4.1 suggests that poles sometimes are created at a higher concentration if the domain contains holes.

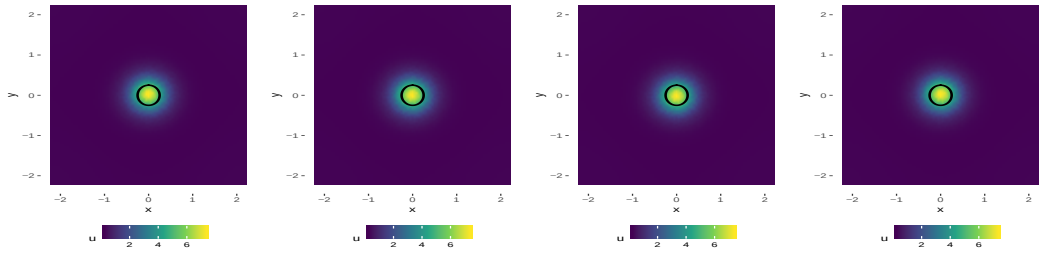
4.2 Effects of controlling the initial steady state

To investigate if the poles could be spatially confined by disturbing the steady state in a controlled manner, five experiments each were run on the domains with zero, five and 20 holes. In each simulation the homogeneous steady state was kept unperturbed everywhere, except in a small circle of radius 0.25 where it was randomly perturbed with a positive noise. As the u-component of the models is self-activating (section 2.2.2) it was speculated that this procedure would result in the formation of a pole in the disturbance zone, similar to what is observed in the one dimensional case (Fig. 2.2).

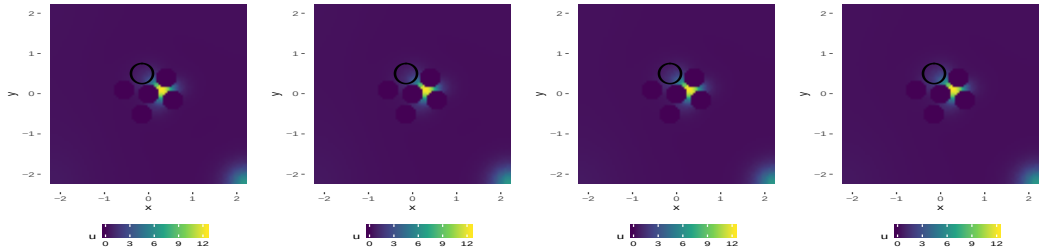
To investigate the result from disturbing the steady state, four out of the five simulations for each domain (zero, five and 20 holes) were plotted at the end time steady state for the Gierer-Meinhardt model (Fig. 4.3). A prominent feature is that within each domain, the same spatial patterns were formed[1] (Fig. 4.3). Another feature is that in the zero holes case, the formed poles overlap with the disturbance region (Fig. 4.3a). However, for the five and 20 holes cases this does not hold, instead the pole is formed a small distance away from the disturbed region (Fig. 4.3b and 4.3c). As this result is rather surprising considering the one dimensional case (Fig. 2.2), the simulations for the five holes domain were rerun using a mesh with $lc = 0.02$ in order to exclude numerical errors as a potential influence. Making the mesh finer did not have any effect (Fig. C.5). For the Schnakenberg model the created poles occupy

[1]This also holds for the other six cases not included in Fig. 4.3 (result not shown here).

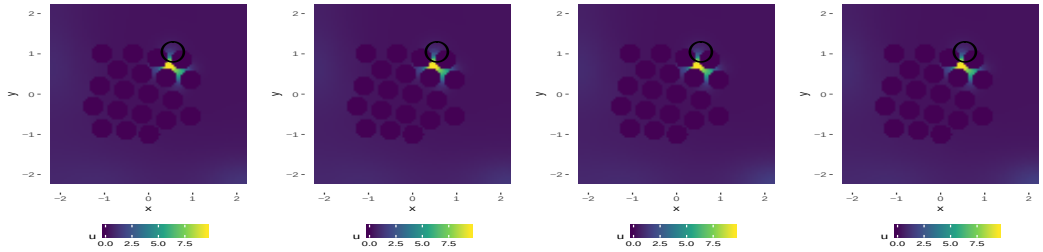
a larger area. Besides this similar results hold (Fig. C.4). That is, the same spatial pattern is formed within each domain and for the five and 20 hole cases the created poles do not have their centre overlapping with the disturbed region. Overall, the result suggests that perturbing the steady state in a small region results in the same spatial pattern being generated. Furthermore, Fig. 4.3 and Fig. C.4 suggest that for domains with many holes, disturbing the steady state fails to spatially confine a pole to a specific region.



(a) *Zero holes*



(b) *Five holes*



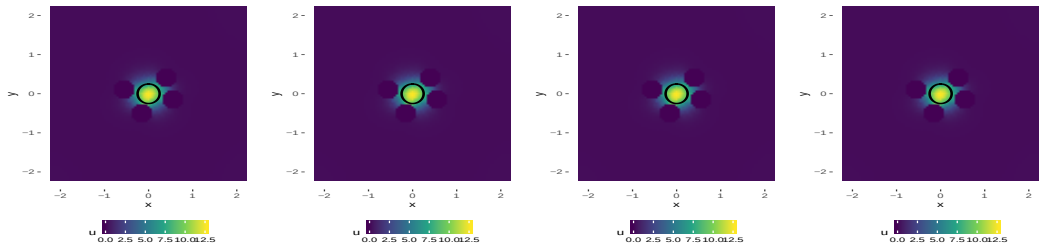
(c) *20 holes*

Figure 4.3: *The u value for zero, five and 20 holes for Gierer-Meinhardt model at* $\tau = 2.0$ *when the steady state was disturbed at a specific region (black circle). The data is generated from four different simulations where the steady state only was perturbed randomly within the black circle.*

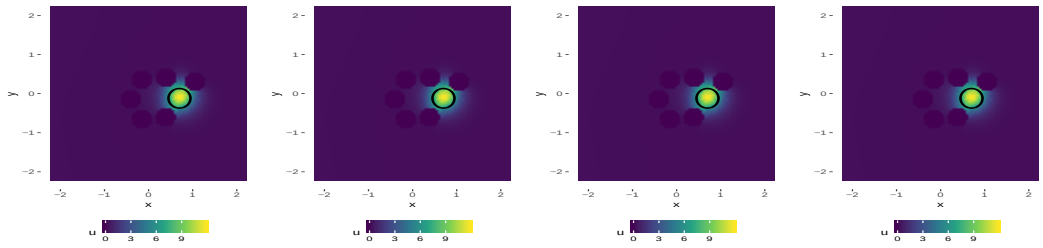
4.3 Effects of different parameters in a sub-region

To investigate if the poles could be spatially confined by using varying parameters, many different experiments were conducted where the parameters had different values within a small circular sub-region of radius = 0.25. The first of these experiments focused on changing the a and d parameters within the Turing space. In these attempts the a-parameter, which dictates the constant production of u, was increased

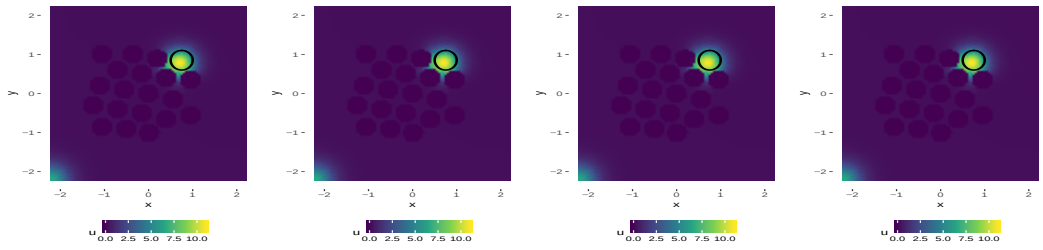
for both models within the limits of the Turing spaces. Based on the intuitive explanation in (section 2.2.2) it was hypothesised that if the self-activating component u had a larger constant production in a small region, a pole should be confined to that region. In addition to increasing a, the relative diffusion d was also increased in some attempts. As the faster diffusion of v is responsible for confinement of u (section 2.2.2), it was hypothesised that a larger diffusion within a sub-region might result in u being more easily confined to that region. Overall, all the attempts of changing a and d within the Turing space for a small circular sub-region yielded, for the five hole domain[2], similar results to the case where the initial steady state was controlled (Fig. 4.3b). That is, the same spatial patterns are created in each run, but the final pole appears at a small distance away from the region with, in this case, a larger a and d value. It should be noted that the b-parameter was not changed in these attempts, as the shape of the Turing space (Fig. 2.3) makes it unfeasible to change b if a is changed.



(a) *Five holes*



(b) *Seven holes*



(c) *20 holes*

Figure 4.4: *The u value for five, seven and 20 holes for Gierer-Meinhardt model at $\tau = 2.0$ when the parameters have values outside the Turing space in a subdomain (black circle). The data is generated from four simulations with different random perturbations of the initial steady state over the entire domain.*

[2]These computationally demanding approaches were only attempted on the five holes domain, in order to allow for more parameter combinations to be attempted.

Considering that changing parameters within the Turing space in a sub-region failed to control the pole formation, it was attempted to instead use parameters outside of the Turing space. The feasibility of this approach has been suggested in the litterature, where in the one dimensional case poles have been spatially confined using parameters outside the Turing space [10]. For the Gierer-Meinhardt model, a was increased and b was decreased outside of the Turing space. The motivation for also changing b, which controls the degradation of u, is the same as for changing a; if more u is produced in a region the chance of a pattern being confined in that region should increase. For the Schnakenberg model similar changes were applied, increasing a and decreasing b. The motivation for decreasing b, which controls constant production of v, is that in the Schnakenberg model an increase in u and a decrease in v are both responsible for creating the pole (section 2.2.2).

For the Gierer-Meinhardt model changing a and b outside of the Turing space resulted in the pole being confined within the desired sub-region for each simulation on the five holes domain (Fig. 4.4a). To ensure that this was not due to the geometry of the domain, the analysis was also performed on a domain with seven and 20 holes. It can be observed that changing the domain did not have any effect on the confinement of the pole to the desired region (Fig. 4.4b and 4.4c). Changing a and b outside the Turing space for Schnakenberg model yielded similar result (Fig. C.6). That is, the pole is centred in the region with different parameters. A difference however, is that the pole of the Schnakenberg model occupies a larger surface area, resulting in the pole not being solely confined to the sub-region. Overall, the result suggests that changing parameters outside the Turing space in a sub-region can be an efficient method for spatially confining the centre of a pole to a certain a region.

5

Discussion

In this thesis, the impact of holes (Fig. 1.1c) on pattern formation was examined for the Gierer Meinhardt model (2.6), and the Schnakenberg model (2.8). Furthermore, approaches for spatially confining poles (regions of high concentration) to a specific region were investigated.

For both models, the presence of small densely packed holes sometimes resulted in the creation of poles with higher concentration (Fig. 4.1). Furthermore, for the five holes domain and especially for the 20 holes domain, poles had a tendency to be created close to, or directly in, the region dense with holes (Fig. 4.2 and C.3). This localisation of the poles might be due to the self-activating component u, especially in the case with many holes, more easily starting to accumulate in the diffusion-restricted region between holes compared to the rest of the domain. In addition, an accumulation in a diffusion restricted region might explain the faster pole creation time and higher pole-concentration for the Gierer-Meinhardt model, since more u is available in a small region for boosting the self-enhancement. Why the pattern creation time did not also decrease for the Schnakenberg model is not known, but it might be due that in the Schnakenberg model u inhabits its activator v (section 2.2.2). Consequently, although more u might be available in small region to sometimes yield a higher pole concentration, the dependency on v slows down the pole creation time. However, it should be noted that this is only speculations based on the result in section 4.1.

Changing parameter values outside the Turing space in a sub-region proved to be the most efficient method for spatially confining poles in both models (Fig. 4.4 and C.6). Although not as successful, disturbing the steady state in a sub-region, or changing the parameters within the Turing space in a sub-region, proved able to confine poles close to that sub-region. Overall, this suggests that different parameters in a sub-region, or a controlled disturbance of the steady state, might be the explanation behind a scenario where a pole always is formed in, or close to, a specific region. Though successful, it should be noted that the action of changing parameters, or controlling the initial disturbance, in a sub-region for a model poses a new question. That is, how did the parameters achieve a different value in a sub-region, or why is the steady state disturbed differently in a sub-region? If these questions cannot be answered, changing the parameters or controlling the initial disturbance will result in a model where the underlying model assumptions cannot be properly motivated.

Although attempts at controlling pole formation worked relatively well, it should be noted that only two models were investigated. Consequently, as future work it

would be relevant to investigate if the approaches outlined in section 4.2 and 4.3 work for a broader class of RD-models, like the Thomas model [4], and/or the Gray-Scott model [18]. Furthermore, as some real-life pattern materialise in systems that are three dimensional, it is of relevance to investigate the impact of holes, and if pole formation can be controlled, for a three dimensional domain. In particular, as the polarisation site in *S. cerevisiae* appear close to holes (budding scars) [9], a spherical surface-bulk RD-system [19], could be of extra interest to investigate. Lastly it might, for both a two and three dimensional domain, be of relevance to look into other hole structures, as small densely packed circular holes probably not are the only holes to appear in biological systems.

If more simulations for a different model or domain are to be carried out, the runtime of the PDE-solver should be improved. Currently, it takes relatively long time to solve the PDE-system (12-20 minutes), and as several simulations are required to understand the general impact of holes on a model or domain, the overall runtime becomes substantial (chapter 3). An improvement in runtime could be achieved by using an inhomogeneous grid with non-equidistant nodes. Currently the mesh is very fine ($lc = 0.04$) over the entire domain, this is in order to control the numerical error in the regions between the holes. It is likely that a courser grid suffices outside the dense hole region, resulting in fewer nodes and thus a faster runtime. A mesh with varying mesh-size is to be especially preferred if a three dimensional domain with holes is considered, as the number of nodes in a mesh increases drastically with an extra dimension.

Besides mesh density, an adaptive time-steeping procedure could improve the run time. Currently, the time-step size Δt is kept at a small constant value in order to control the numerical error, however it is questionable if small time-steps are required at each time point. It is probable that runtime could be improved by an adaptive solver that takes large time steps in regions where a lot changes in the system, but small time-steps when the system barely changes. Besides improving runtime, such an adaptive step-length would remove the need of fine-tuning the step-length for different models or domains. To obtain a suitable adaptive step-length, the errors (residuals) obtained from solving the linear system that Eq. (3.5) gives rise to could be used. Overall, if the mesh is made adaptive and an adaptive time-stepping algorithm is implemented, it becomes more feasible to investigate the impact of holes on other models and/or more realistic domains.

To conclude, the result suggests that small densely packed holes have an effect on pattern formation (Fig. 4.2 and C.3). That is, on a domain with many holes, poles have a tendency to accumulate close to, or directly in, the region dense in holes. The exact reason behind this behaviour is not known, but it might be due to u being confined to the diffusion-restricted region between holes. Regarding controlling pole formation, changing parameters outside the Turing space in a sub-region proved efficient for spatially confining poles to a specific region (Fig. 4.4 and C.6). Although potentially useful for recreating observed patterns, it is important to note that this method raises a new question; why are the parameter values different in a sub-region? Lastly, it should be noted that this thesis also has demonstrated FEM as a useful method for solving RD-systems on domains with complex geometries.

Bibliography

[1] Kondo S, Miura T. Reaction-diffusion model as a framework for understanding biological pattern formation. science. 2010;329(5999):1616–1620.

[2] Wedlich-Soldner R, Altschuler S, Wu L, Li R. Spontaneous cell polarization through actomyosin-based delivery of the Cdc42 GTPase. Science. 2003;299(5610):1231–1235.

[3] Meinhardt H. Models of biological pattern formation. New York. 1982;.

[4] Murray JD. Mathematical biology. II Spatial models and biomedical applications. Springer; 2003.

[5] Raspopovic J, Marcon L, Russo L, Sharpe J. Digit patterning is controlled by a Bmp-Sox9-Wnt Turing network modulated by morphogen gradients. Science. 2014;345(6196):566–570.

[6] Turing AM. The chemical basis of morphogenesis. Bulletin of mathematical biology. 1990;52(1-2):153–197.

[7] Schnakenberg J. Simple chemical reaction systems with limit cycle behaviour. Journal of theoretical biology. 1979;81(3):389–400.

[8] Gierer A, Meinhardt H. A theory of biological pattern formation. Kybernetik. 1972;12(1):30–39.

[9] Meitinger F, Khmelinskii A, Morlot S, Kurtulmus B, Palani S, Andres-Pons A, et al. A memory system of negative polarity cues prevents replicative aging. Cell. 2014;159(5):1056–1069.

[10] Page K, Maini PK, Monk NA. Pattern formation in spatially heterogeneous Turing reaction–diffusion models. Physica D: Nonlinear Phenomena. 2003;181(1-2):80–101.

[11] Page KM, Maini PK, Monk NA. Complex pattern formation in reaction–diffusion systems with spatially varying parameters. Physica D: Nonlinear Phenomena. 2005;202(1-2):95–115.

[12] Geuzaine C, Remacle JF. Gmsh: A 3-D finite element mesh generator with built-in pre-and post-processing facilities. International journal for numerical methods in engineering. 2009;79(11):1309–1331.

[13] Logg A, Mardal KA, Wells G. Automated solution of differential equations by the finite element method: The FEniCS book. vol. 84. Springer Science &

Business Media; 2012.

[14] Noble WS. A quick guide to organizing computational biology projects. PLoS computational biology. 2009;5(7):e1000424.

[15] Wilson G, Bryan J, Cranston K, Kitzes J, Nederbragt L, Teal TK. Good enough practices in scientific computing. PLoS computational biology. 2017;13(6):e1005510.

[16] Larsson S, Thomée V. Partial differential equations with numerical methods. vol. 45. Springer Science & Business Media; 2008.

[17] Bitterman KJ, Medvedik O, Sinclair DA. Longevity regulation in Saccharomyces cerevisiae: linking metabolism, genome stability, and heterochromatin. Microbiol Mol Biol Rev. 2003;67(3):376–399.

[18] Gray P, Scott S. Autocatalytic reactions in the isothermal, continuous stirred tank reactor: Oscillations and instabilities in the system A + 2B $\rightarrow$ 3B; B $\rightarrow$ C. Chemical Engineering Science. 1984;39(6):1087–1097.

[19] Rätz A, Röger M. Symmetry breaking in a bulk–surface reaction–diffusion model for signalling networks. Nonlinearity. 2014;27(8):1805.

[20] Teschl G. Ordinary differential equations and dynamical systems. vol. 140. American Mathematical Soc.; 2012.

A

Solution for a linear ODE-system

Assume a linear ODE-system on the form:

$$\frac{\mathrm{d}(\delta \mathbf{x})}{\mathrm{d}t} = \mathbf{J}^* \delta \mathbf{x} \tag{A.1}$$

Where $\mathbf{J}^*$ is the Jacobian matrix evaluated at the steady state. The general solution to the system in Eq. (A.1) is [20]:

$$\delta \mathbf{x}(t) = \delta \mathbf{x}_0 \exp(t\mathbf{J}^*) \tag{A.2}$$

Where the matrix exponential is defined as:

$$\exp(t\mathbf{J}^*) = \sum_{j=0}^{\infty} \frac{t^j}{j!} (\mathbf{J}^*)^j \tag{A.3}$$

Evaluating the matrix exponential in Eq. (A.3) is generally a non trivial task. However there are special cases where it can easily be evaluated. The one that will be discussed here, and also the one that holds for $\mathbf{J}^*$ in section 2.2, is when $\mathbf{J}^*$ is diagonalizable. This means that $\mathbf{J}^*$ has n independent eigenvectors which results in the matrix being expressible as $\mathbf{J}^* = \mathbf{U}\mathbf{D}\mathbf{U}^{-1}$, where the columns of $\mathbf{U}$ are linearly independent eigenvectors and $\mathbf{D}$ is a diagonal matrix with the corresponding eigenvalues. Given this, the matrix exponential in Eq. (A.3) can be evaluated by first noting that:

$$(\mathbf{J}^*)^j = (\mathbf{U}\mathbf{D}\mathbf{U}^{-1})^j = (\mathbf{U}\mathbf{D}\mathbf{U}^{-1})(\mathbf{U}\mathbf{D}\mathbf{U}^{-1}) \cdots (\mathbf{U}\mathbf{D}\mathbf{U}^{-1}) = \mathbf{U}\mathbf{D}^j\mathbf{U}^{-1} \tag{A.4}$$

Inserting this into the matrix exponential results in:

$$\exp(t\mathbf{J}^*) = \sum_{j=0}^{\infty} \mathbf{U} \frac{\mathbf{D}^j t^j}{j!} \mathbf{U}^{-1} = \mathbf{U}\exp(\mathbf{D}t)\mathbf{U}^{-1} \tag{A.5}$$

As $\mathbf{D}$ is a diagonal matrix applying the matrix exponential simply results in:

$$\exp(\mathbf{D}t) = \begin{bmatrix} e^{\lambda_1 t} & 0 & \cdots & 0 \\ 0 & e^{\lambda_2 t} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & e^{\lambda_n t} \end{bmatrix} \tag{A.6}$$

So overall the general solution becomes:

$$\delta\mathbf{x}(t) = \mathbf{U}\exp(\mathbf{D}t)\mathbf{U}^{-1}\mathbf{x}_0 = \sum_{k=1}^{n} c_k \mathbf{u}_k e^{\lambda_k t} \tag{A.7}$$

Where λ_k is the eigenvalues of the matrix $\mathbf{J}^*$ and $\mathbf{u}_k$ are the eigenvectors.

B

Parameter values for plots

The parameter values used for each plot in this report are presented in Tab. B.1.

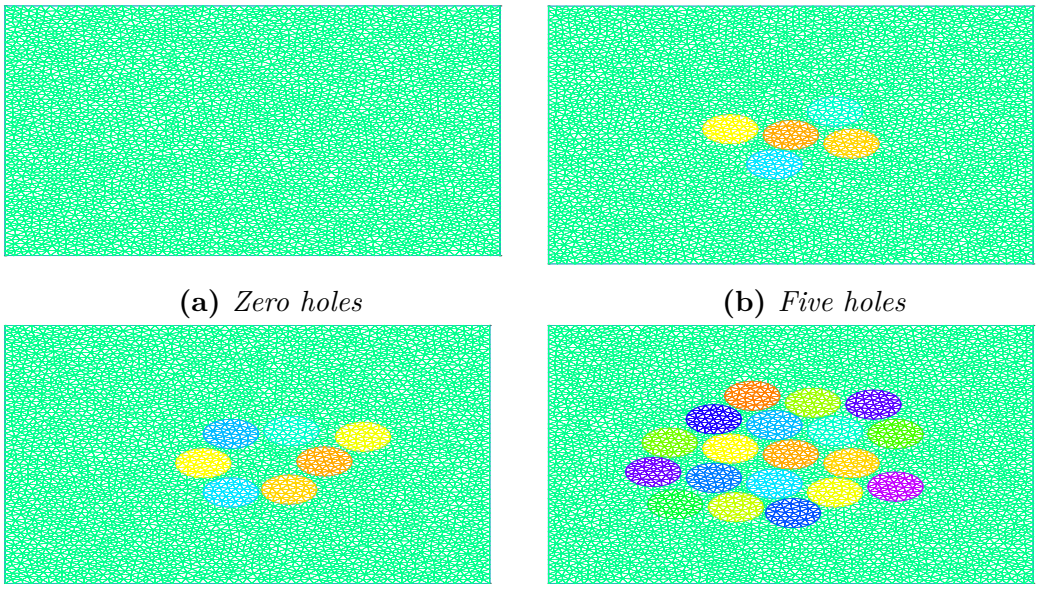
Table B.1: *Parameter values used for each plot presented in this report.*

Figure	Parameters big domain				Parameters subdomain			
	a	b	γ	d	a	b	γ	d
1.1a	0.2	2.0	10.0	100.0	-	-	-	-
1.1b	0.5	2.0	20.0	50.0	-	-	-	-
2.2a	0.5	2.0	20.0	50.0	-	-	-	-
2.2b	0.5	2.0	5.0	200.0	-	-	-	-
4.2(a-c)	0.5	2.0	20.0	50.0	-	-	-	-
4.3(a-c)	0.5	2.0	20.0	50.0	-	-	-	-
4.4(a-c)	0.5	2.0	20.0	50.0	2.0	0.5	20.0	50.0
C.3(a-c)	0.2	2.0	10.0	100.0	-	-	-	-
C.4	0.2	2.0	10.0	100.0	-	-	-	-
C.6	0.2	2.0	10.0	100.0	2.5	0.2	10.0	100.0

C

Supplementary figures

C.1 Finite-element meshes



(a) *Zero holes*

(b) *Five holes*

(c) *Seven holes*

(d) *20 holes*

Figure C.1: *The finite element meshes. Each mesh has a* $lc = 0.08$ *for illustrative purposes and only the rectangular base-domain meshes are displayed, as the circular base-domain have the exact same hole-structure.*

C.2 Effect of holes

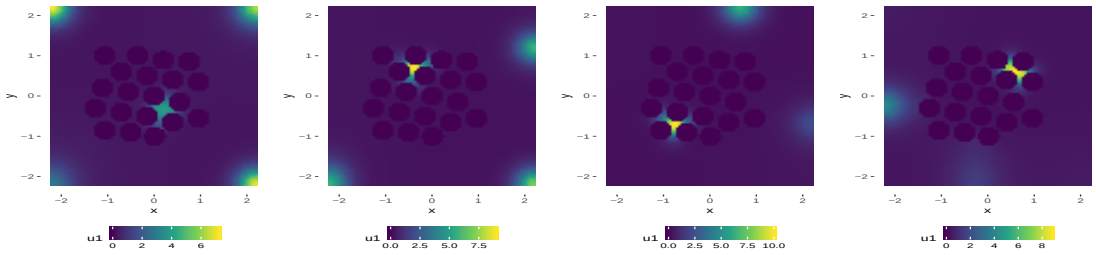
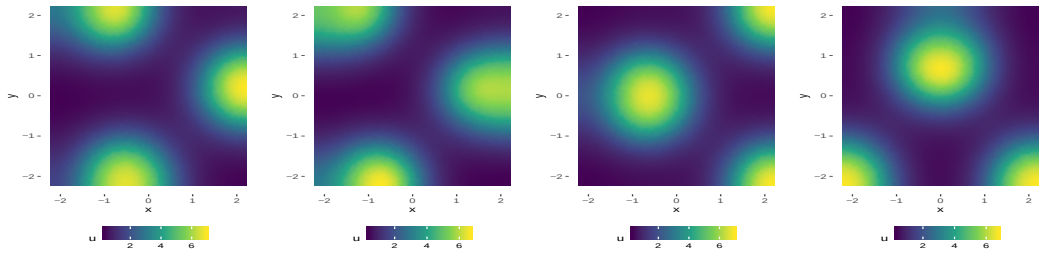
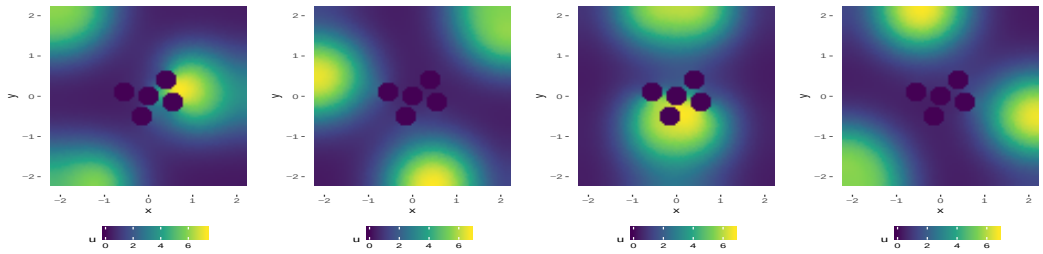


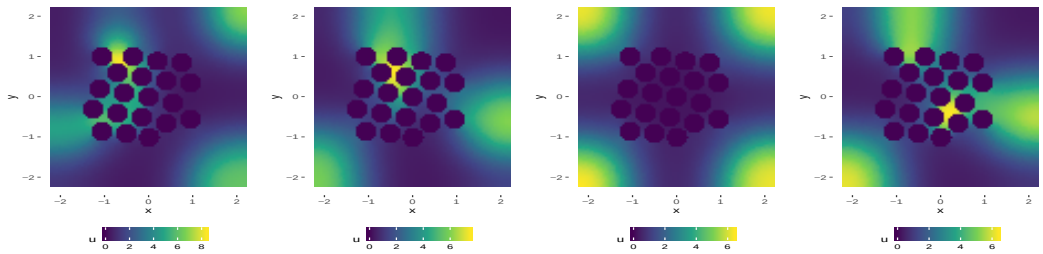
Figure C.2: *The u value for 20 holes for the Gierer-Meinhardt model at $\tau = 2.0$. The data is generated from four simulations with different random perturbations compared to those in Fig. 4.2c of the initial steady state.*



(a) *Zero holes*



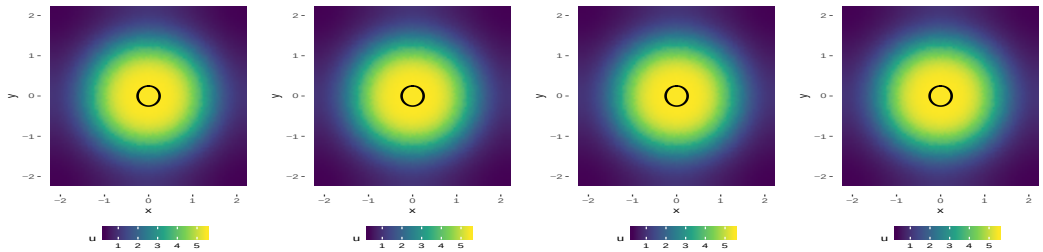
(b) *Five holes*



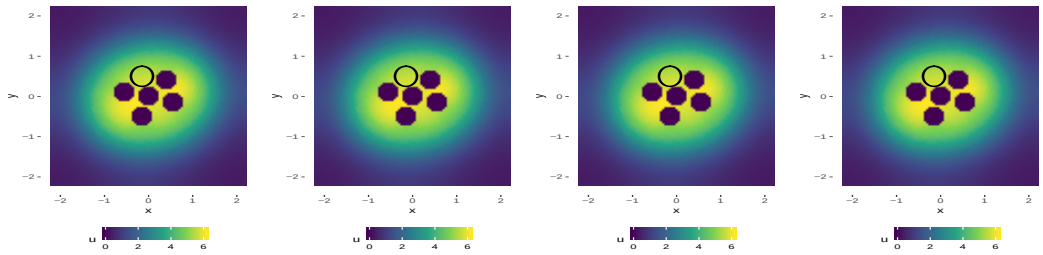
(c) *20 holes*

Figure C.3: *The u value for zero, five and 20 holes for the Schnakenberg model at $\tau = 7.5$. The data is generated from four simulations with different random perturbations of the initial steady state.*

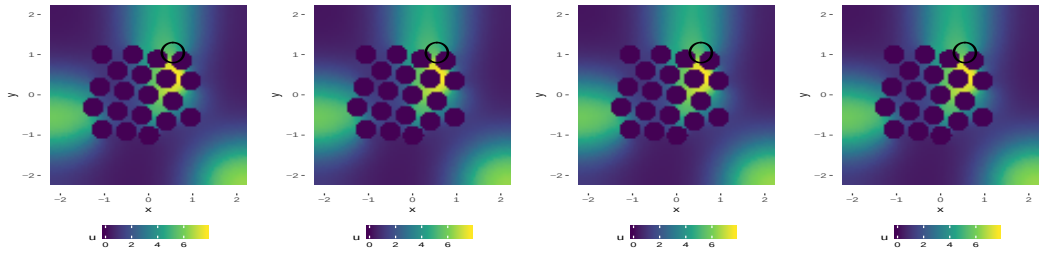
C.3 Effect of controlling initial steady state



(a) *Zero holes*



(b) *Five holes*



(c) *20 holes*

Figure C.4: *The u value for zero, five and 20 holes for the Schnakenberg model at* $\tau = 7.5$ *when the steady state was disturbed at a specific region (black circle). The data is generated from four different simulations where the steady state only was perturbed randomly within the black circle.*

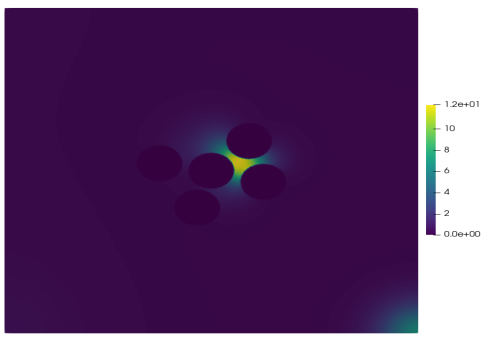
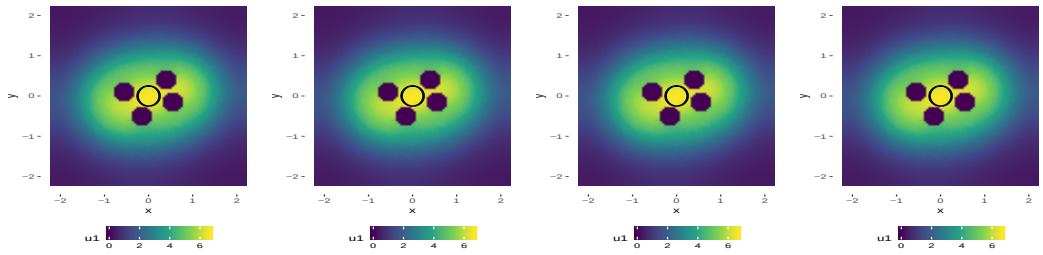
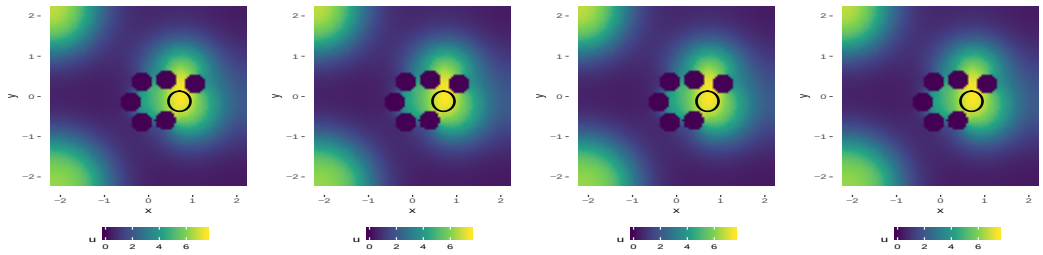


Figure C.5: *Controlling the initial steady state with a fine mesh (lc = 0.02) for the Gierer-Meinhardt model. The data is generated the same way as in Fig. 4.3.*

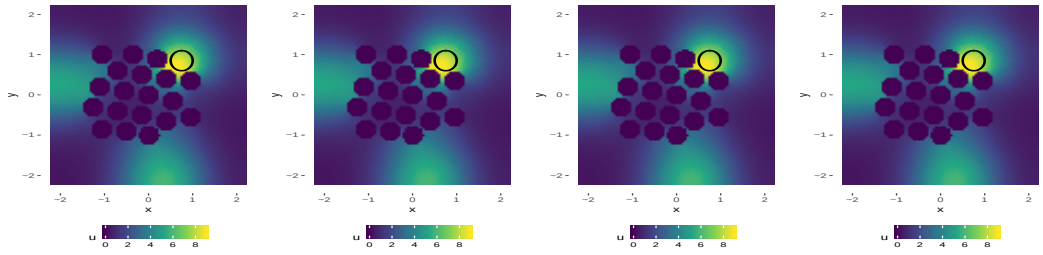
C.4 Effect of different parameters in subregion



(a) *Five holes*



(b) *Seven holes*



(c) *20 holes*

Figure C.6: *The u value for five, seven and 20 holes for the Schnakenberg model at $\tau = 7.5$ when the parameters have values outside the Turing-space in a subdomain (black circle). The data is generated from four simulations with different random perturbations of the initial steady state over the entire domain.*