

PERFECT STRIPES FROM A GENERAL TURING MODEL
IN DIFFERENT GEOMETRIES

by

Jean Tyson Schneider

A thesis

submitted in partial fulfillment
of the requirements for the degree of
Master of Science in Mathematics

Boise State University

August 2012

© 2012
Jean Tyson Schneider
ALL RIGHTS RESERVED

BOISE STATE UNIVERSITY GRADUATE COLLEGE

DEFENSE COMMITTEE AND FINAL READING APPROVALS

of the thesis submitted by

Jean Tyson Schneider

Thesis Title: Perfect Stripes from a General Turing Model in Different Geometries

Date of Final Oral Examination: 03 May 2012

The following individuals read and discussed the thesis submitted by student Jean Tyson Schneider, and they evaluated her presentation and response to questions during the final oral examination. They found that the student passed the final oral examination.

Donna Calhoun, Ph.D.

Chair, Supervisory Committee

Jodi Mead, Ph.D.

Member, Supervisory Committee

Grady Wright, Ph.D.

Member, Supervisory Committee

The final reading approval of the thesis was granted by Donna Calhoun, Ph.D., Chair, Supervisory Committee. The thesis was approved for the Graduate College by John R. Pelton, Ph.D., Dean of the Graduate College.

This thesis is dedicated to Loretta Belgum, Ph.D., Aunt, mentor, and prayer warrior.

ACKNOWLEDGMENTS

I wish to express my sincere appreciation to the following people:

- Dr. Alan Hausrath, for encouraging me to pursue a master's degree in mathematics.
- Dr. Barbara Zubik-Kowal, for encouraging me to write a thesis.
- Dr. Jodi Mead, for preparing me to write a thesis.
- Dr. Donna Calhoun, for overseeing this thesis and sharing her extensive knowledge in such a positive environment.
- April, Anne, Joe, and John Schneider for making due without the full attention of their mother.
- Mike Schneider, my husband, for supporting me in this academic endeavor.

ABSTRACT

We explore a striped pattern generated by a general Turing model in three different geometries. We look at the square, disk, and hemisphere and make connections between the stripes in each spatial direction. In particular, we gain a greater understanding of when perfect stripes can be generated and what causes defects in their patterns. In this investigation, we look at the difference between the solutions due to the different domain shapes. In the end, we propose a reason why stripes from a reaction-diffusion system with zero-flux boundary conditions can be perfect on a square or hemisphere, but not on a disk.

TABLE OF CONTENTS

ABSTRACT	vi
LIST OF TABLES	ix
LIST OF FIGURES	x
1 INTRODUCTION	1
2 DIFFUSION-DRIVEN INSTABILITY	3
2.1 A Picture of Diffusion-Driven Instability	3
2.2 Linear Stability Analysis	6
3 TURING MODELS	11
3.1 Common Turing Models	11
3.2 A General Turing Model	12
3.3 Parameter Selection	15
4 NUMERICAL SIMULATIONS	17
4.1 The Dispersion Relation and a Pattern in 1-D	17
4.2 Square Domain	19
4.3 Circular Domain	24
4.3.1 Analysis of the BVAM Model on a Disk	24
4.3.2 Numerical Method and Results	26

4.4	Hemisphere	31
5	CONCLUSIONS	33
5.1	Perfect Stripes in Different Geometries	33
5.2	Exploring Stripes on Other Domains	35
	REFERENCES	37
A	MATLAB CODE	38
A.1	One-Dimension Code for Square Domain	38
A.2	Two-Dimension Code for Square Domain	40
B	TABLES AND FIGURES FROM NUMERICAL SIMULATIONS	43
B.1	Square	43
B.2	Disk	44
B.3	Hemisphere	45

LIST OF TABLES

5.1	Modal comparison by geometry	34
5.2	Ratio between stripes across the disk and waves around the disk	35
B.1	Values of k and δ based on the number of stripes	43
B.2	Values which correspond to Figure B.1	44
B.3	Fourier-Bessel coefficients from $k = 17$ in Figure 4.10	46
B.4	Number of stripes on a hemisphere given the k value	47

LIST OF FIGURES

2.1	Diffusion of species U without reaction kinetics for various times	4
2.2	Reaction kinetics without diffusion for various times	5
2.3	Diffusion-driven instability for various times	5
2.4	Comparison of activator and inhibitor at steady-state	6
4.1	The dispersion relation and Turing patterns in one dimension	18
4.2	The two dimensional BVAM model in a square domain with 5 waves (stripes) on the left and 5.25 waves (stripes) on the right.	21
4.3	The result of integer number of stripes on the left and half-integer number of stripes on the right.	22
4.4	Predicted number of waves in each direction: 4.5 stripes on the left and 6 stripes on the right.	22
4.5	Competition between horizontal waves and vertical waves can cause defects.	23
4.6	Perfect Stripes	24
4.7	Solution on a disk and its reconstructed form from Fourier-Bessel coefficients where $q(1)$ corresponds to activator species U	26
4.8	Stripes on a disk for $k = 10$, $m = 8$ on the left and $k = 12$, $m = 10$ on the right	28
4.9	Fourier-Bessel coefficients for $k = 10$ on the left and $k = 12$ on the right	29
4.10	The solution and the corresponding Fourier-Bessel coefficients for $k = 17$	30

4.11	Solution for $k = 18$	31
4.12	Hemisphere with $k = 8$ and $k = 11$	32
4.13	Hemisphere with slight defect	32
B.1	Results selecting k as an integer from $k = 10$ to $k = 18$	45
B.2	Hemisphere with $k = 8, 9, 10$ in row 1, $k = 11, 12, 13$ in row 2, and $k = 14, 15a, 15b$ in row 3.	47

CHAPTER 1

INTRODUCTION

A reaction-diffusion model is a system of partial differential equations that describe the evolutionary change in concentration of two or more chemicals in a solution. It was first suggested by Alan Turing in 1952 [7] that while reaction-diffusion equations generally describe a stable process leading to a homogeneous solution, under certain circumstances an instability could occur leading instead to an inhomogeneous solution. If one could see the different levels of concentration that developed in the solution, they would note a pattern had formed. Because chemical reactions are often very fast, it was difficult to demonstrate, experimentally, the patterns in solution that Turing's mathematical model had demonstrated. As a result, not much attention was paid to Turing's ideas about diffusion-driven instabilities until the late 1970s and into the 1980s when his ideas were revived. Over the course of that decade and into the 1990s, chemists were able to show pattern formations in an actual Turing-type chemical reaction. Maini et al. report, in their overview of the research done on spatial patterns in chemical and biological systems, that De Kepper and group were first to present experimental evidence of Turing patterns [4]. During the 80s and 90s, many scholars from chemistry to biology to mathematics studied this phenomenon both in the lab and through numerical simulation. Maini et al. have written a complete review of the work that was done in both chemistry and biology over this

period of time [4]. The application of Turing structures was not limited to just these fields, but the ideas spread to many areas of science including ecology, semiconductor physics, material science, hydrodynamics, astrophysics, and economics.

In 2009, Calhoun and Helzel [3] developed a finite volume method for solving parabolic equations on curved surfaces. They demonstrated their technique on a Turing model applying the pattern formation to different surfaces. While their numerical method was robust, it was noticed that the same model could generate clear stripes on a hemisphere, but show defects in the stripes when applied to a circular disk or a sphere. Much of the research of Turing models has focused on different model types, the kinds of patterns formed, and their application to the development of patterned structures in living organisms. Controlling the pattern formation has received less attention. Sanderson et al. [6] addressed this issue for two types of Turing models from the perspective of texture synthesis and scientific computing.

In this thesis, we show how to choose the number of stripes formed in various geometries using a general Turing model. We will also look at how the striped patterns interact in rectangular and circular domains. First, we will discuss how an instability can occur in a reaction-diffusion system, then we will briefly look at several Turing models before introducing a specific model and discussing our parameter selections. We will present our numerical methods and show the results of our numerical simulations. Through this process, we propose an explanation why perfect stripes can be produced on a square grid or on a hemisphere, but not on a disk.

CHAPTER 2

DIFFUSION-DRIVEN INSTABILITY

2.1 A Picture of Diffusion-Driven Instability

Diffusion describes the movement of particles from an area of higher concentration to an area of lower concentration to bring about equilibrium. It is normally thought of as a smooth and stable process. When Alan Turing [7] suggested that a solution of two or more chemicals in a homogeneous state could become inhomogeneous specifically because of diffusion, it was a revelation. Turing showed that if the chemicals worked in opposition to each other, for example, in a growth and decay relationship or an activator-inhibitor relationship, and if the reaction rate of the decay or inhibitor agent is significantly faster than the growth or activator rate, then this homogeneous state would become inhomogeneous because the diffusion would carry this asymmetry throughout the solution. This is what Turing coined as “diffusion-driven instability [7].” While most reaction-diffusion models are stable, when the kinetics of the reactions fall within a narrow parameter regime, these instabilities cause the formation of patterned concentrations.

The mathematical model Turing proposed was

$$\frac{\partial \mathbf{U}}{\partial t} = D \nabla^2 \mathbf{U} + F(\mathbf{U}) \quad (2.1)$$

where $\mathbf{U} = \mathbf{U}(\mathbf{x}, t) \in \mathfrak{R}^n$ represents the concentrations of n chemicals in space at a particular time, and D is a diagonal matrix of diffusion coefficients for each chemical. The chemical reactions are modeled by the function $F(\mathbf{U})$, which is usually non-linear. This system would have as many equations as chemicals in the solution. Typically, to keep the model simple, we consider a system of two chemicals, U and V , on a bounded domain with Neumann, Dirichlet, or periodic boundary conditions. Rewriting Equation (2.1) as a coupled system gives

$$\begin{aligned} U_t &= D_U \nabla^2 U + f(U, V) \\ V_t &= D_V \nabla^2 V + g(U, V). \end{aligned} \tag{2.2}$$

We begin by giving some meaning to the above equation. The partial derivatives, U_t and V_t , express the change in the concentration of the chemicals over time. The functions f and g describe the reaction kinetics taking place in the solution and $D_U \nabla^2 U$ and $D_V \nabla^2 V$ model the diffusion process. If there are no reaction kinetics (i.e., no f and g), then we have a diffusion equation that does not display an instability. The behavior of this system is demonstrated in one dimension in Figure 2.1. The initial

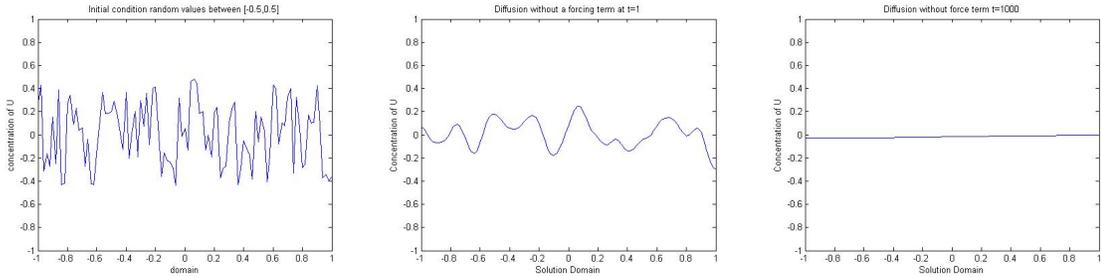


Figure 2.1: Diffusion of species U without reaction kinetics for various times

conditions are random values between $[-0.5, 0.5]$ where these values are considered as deviations about a homogeneous state, to be made precise later. Almost immediately,

the concentration diffuses to a smooth, uniform solution. Alternatively, we can look at the behavior of the levels of concentration in the absence of diffusion. In this case, we have $D = 0$, and there is no transport of the chemicals in the domain. In Figure 2.2, we show the concentration levels for a system in one dimension. In particular, we notice that the evolution of the system is not smooth.

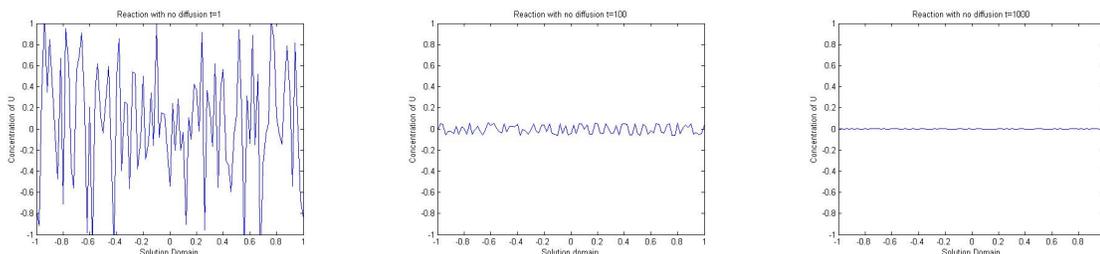


Figure 2.2: Reaction kinetics without diffusion for various times

Finally, we show that for a particular choice of parameters, we can get an instability with diffusion. In Figure 2.3, we see that the solution reaches a final steady-state in which the concentration varies in the spatial domain, revealing a specific smooth pattern. This is the result of a diffusion-driven instability. This type of instability

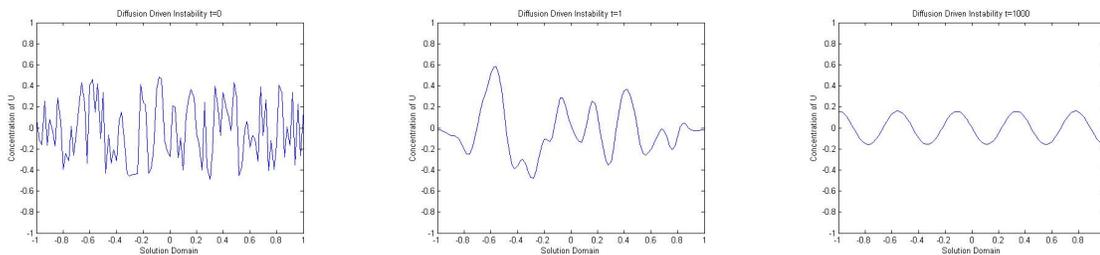


Figure 2.3: Diffusion-driven instability for various times

relies on species working in opposition to each other. While we have focused our attention in these diagrams on the activator or growth element, U , the opposition chemical or inhibitor, V , will always exhibit similar behavior. In Figure 2.4, we show both U and V as a result of diffusion-driven instability. While the level of

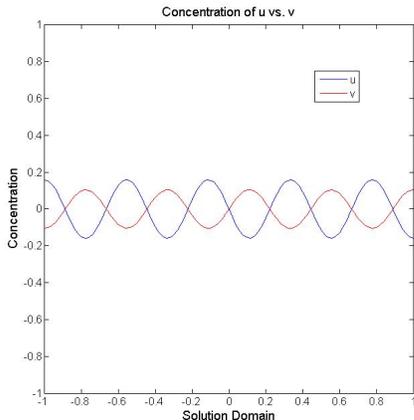


Figure 2.4: Comparison of activator and inhibitor at steady-state

concentrations are not exactly opposite, the maximums of one occur at the minimums of the other. It was noted above that only a narrow set of parameters could bring about diffusion-driven instability in a reaction-diffusion model. In the next section, we look at the analysis that defines this set of parameters.

2.2 Linear Stability Analysis

Turing reaction-diffusion models are generally non-linear. As such, it can be difficult to understand how a particular solution will develop over time and space. We can gain some understanding of the behavior of the solution by looking at one solution over time. We first linearize the reaction function about the homogeneous steady-state solution. Then, the linear stability analysis looks at the time component of a particular solution to see what growth rates will converge to zero, producing a stable-state much as we saw in Figure 2.1. From this, we can look at the conditions for which instabilities can occur. For simplicity, we carry out the analysis in one spatial dimension.

The uniform stationary state solution (U_0, V_0) satisfies $f(U_0, V_0) = g(U_0, V_0) = 0$. Using a Taylor expansion about (U_0, V_0) , we find a linear approximation of f and g to be

$$\begin{aligned} f(U, V) &\approx f(U_0, V_0) + f_U(U_0, V_0)(U - U_0) + f_V(U_0, V_0)(V - V_0) \\ g(U, V) &\approx g(U_0, V_0) + g_U(U_0, V_0)(U - U_0) + g_V(U_0, V_0)(V - V_0). \end{aligned} \quad (2.3)$$

Since f and g are both zero at the steady-state, we can rewrite Equation (2.3) in vector form for simplicity. Let $u = U - U_0$ and $v = V - V_0$, then $\mathbf{w} = [u, v]^T$. The partial derivatives of f and g with respect to U and V at (U_0, V_0) can be collected into a matrix, which we will call A . The linear approximation of the reaction function is then $A\mathbf{w}$. From this, we can rewrite the linearized reaction-diffusion system as

$$\frac{\partial \mathbf{w}}{\partial t} = D\nabla^2 \mathbf{w} + A\mathbf{w}, \quad A = \begin{bmatrix} f_U & f_V \\ g_U & g_V \end{bmatrix}. \quad (2.4)$$

We now seek a particular solution corresponding to wavenumber k in the form

$$\mathbf{w}_k = \mathbf{C}_k e^{\lambda_k t} e^{ikx} \quad (2.5)$$

where $Re(\lambda_k)$ is the growth rate and k is the wavenumber of our particular solution. The vector, $\mathbf{C}_k \in \mathfrak{R}^{2 \times 1}$, is the Fourier constants for the system. Substituting this solution into Equation (2.4) gives

$$\frac{\partial}{\partial t} \mathbf{C}_k e^{\lambda_k t} e^{ikx} = D \frac{\partial^2}{\partial x^2} \mathbf{C}_k e^{\lambda_k t} e^{ikx} + A \mathbf{C}_k e^{\lambda_k t} e^{ikx}.$$

or

$$\lambda_k \mathbf{C}_k e^{\lambda_k t} e^{ikx} = -k^2 D \mathbf{C}_k e^{\lambda_k t} e^{ikx} + A \mathbf{C}_k e^{\lambda_k t} e^{ikx}.$$

This leads to the expression $\lambda_k I = -Dk^2 + A$ where I is the identity matrix. From here on, we will let the matrix, A_k , be defined as

$$A_k = A - Dk^2 = \begin{bmatrix} f_U - D_U k^2 & f_V \\ g_U & g_V - D_V k^2 \end{bmatrix}.$$

Thus, our eigenvalue problem is $(A_k - \lambda_k I)\mathbf{w} = 0$, which leads to the characteristic equation

$$\lambda_k^2 - \text{trace}(A_k)\lambda_k + \det(A_k) = 0. \quad (2.6)$$

This can also be expressed as

$$\lambda_k^2 - (f_U + g_V - k^2(D_U + D_V))\lambda_k + (f_U - D_U k^2)(g_V - D_V k^2) - f_V g_U = 0.$$

This implicitly defines λ_k as a function of k^2 . This dispersion relation is discussed further in Section 4.1. For each wavenumber k , there will be a λ_k^1 and λ_k^2 satisfying this equation. The solution will be stable when the growth rate is negative. Therefore, when both $\lambda_k^1 < 0$ and $\lambda_k^2 < 0$, the solution will remain stable. From Equation (2.6) then, we can conclude

$$\text{trace}(A_k) = f_U + g_V - k^2(D_U + D_V) < 0 \quad (2.7)$$

$$\det(A_k) = (f_U - D_U k^2)(g_V - D_V k^2) - f_V g_U > 0. \quad (2.8)$$

Diffusion-driven instability refers to the situation where the solution is stable in the absence of diffusion, but unstable in the presence of diffusion. Setting $D_U = D_V = 0$,

the initial conditions for diffusion-driven instability become clear:

$$\text{trace}(A) = f_U + g_V < 0 \quad (2.9)$$

$$\det(A) = f_U g_V - f_V g_U > 0. \quad (2.10)$$

How can the solution become unstable in the presence of diffusion? In light of Equation (2.9) and the fact that the diffusion coefficients are always positive, Equation (2.7) will always be negative. So, for instability to occur, we must look at Equation (2.8). That is, when is Equation (2.8) negative while maintaining Equation (2.10)?

$$\det(A_k) = D_U D_V k^4 - (f_U D_V + g_V D_U) k^2 + f_U g_V - f_V g_U < 0$$

This is quadratic in k^2 . It is a parabola that opens up since $D_U D_V$ is positive. To find the minimum value of the $\det A_k$, we set the derivative with respect to k^2 equal to zero:

$$\frac{d}{d(k^2)}(\det A_k) = 2D_U D_V k^2 - (f_U D_V + g_V D_U) = 0$$

or

$$k^2 = \frac{f_U D_V + g_V D_U}{2D_U D_V}. \quad (2.11)$$

To find the minimum value, substitute this expression for k^2 into the $\det A_k$. This is the third condition required for diffusion-driven instability to occur.

$$D_U D_V \left(\frac{f_U D_V + g_V D_U}{2D_U D_V} \right)^2 - (f_U D_V + g_V D_U) \left(\frac{f_U D_V + g_V D_U}{2D_U D_V} \right) + f_U g_V - f_V g_U < 0$$

$$D_U D_V \left(\frac{f_U D_V + g_V D_U}{2D_U D_V} \right)^2 - 2D_U D_V \left(\frac{f_U D_V + g_V D_U}{2D_U D_V} \right)^2 + f_U g_V - f_V g_U < 0$$

$$-\frac{(f_U D_V + g_V D_U)^2}{4D_U D_V} + f_U g_V - f_V g_U < 0$$

$$(f_U D_V + g_V D_U)^2 > 4D_U D_V (f_U g_V - f_V g_U)$$

Because of Equation (2.10), it follows that

$$f_U D_V + g_V D_U > 2\sqrt{D_U D_V (f_U g_V - f_V g_U)}.$$

This further implies

$$f_U D_V + g_V D_U > 0,$$

which also follows from Equation (2.11). For clarity, we restate all four conditions here for which diffusion-driven instability will occur:

$$\text{Condition I: } \quad \text{trace}(A) = f_U + g_V < 0 \quad (2.12)$$

$$\text{Condition II: } \quad \det(A) = f_U g_V - f_V g_U > 0 \quad (2.13)$$

$$\text{Condition III: } \quad f_U D_V + g_V D_U > 0 \quad (2.14)$$

$$\text{Condition IV: } \quad f_U D_V + g_V D_U > 2\sqrt{D_U D_V (f_U g_V - f_V g_U)}. \quad (2.15)$$

CHAPTER 3

TURING MODELS

3.1 Common Turing Models

When Turing presented his study of diffusion-driven instability, it was in support of his hypothesis of the existence of “morphogens.” This was a term he coined to represent the idea of shape development. When he demonstrated that a mathematical model of a chemical reaction could produce patterns of stripes and spots, it led him to wonder if this could be the basis for morphogenesis [7]. Since then, a number of biologists have studied reaction-diffusion systems in an attempt to understand the connection between patterns observed in life forms and those that can be produced mathematically through computer simulation. Chaplain et al. [9] used the Schnakenberg system to simulate pattern formation on a sphere. They were able to identify specific chemicals that are produced by tumor cells and are also known to produce growth and inhibit growth in a chemical system. Through their computer simulations, they were able to demonstrate the plausibility of pattern development in tumor growth modeled as a reaction-diffusion system. The Schnakenberg model is shown here as an example of a Turing model that has been studied to gain greater understanding of pattern development in biological systems. This model is

$$\begin{aligned}u_t &= \nabla^2 u + \gamma(a - u + u^2 v), \\v_t &= d\nabla^2 v + \gamma(b - u^2 v),\end{aligned}$$

where the parameters a , b , d , γ are all positive.

Leppänen et al. used the Gray-Scott Model computed in three dimensions to study the inductive signaling mechanism for neuronal growth. This model has been studied extensively and numerous animated demonstrations can be found quickly through the internet. The Gray-Scott model is given by

$$\begin{aligned}u_t &= D_u \nabla^2 u - uv^2 + F(1 - u), \\v_t &= D_v \nabla^2 v + uv^2 - (F + K)v.\end{aligned}$$

In looking at Equation (2.2), one can see the basic model shown in the above systems. In particular, note the linear terms and the cubic terms in each model. The reaction functions are typically non-linear and vary by model in their non-linearity. Some display an inverse non-linearity, such as the CIMA reaction discussed by Maini et al. [4], while others have quadratic or cubic terms.

Biologists have developed a generic Turing model that has been used to imitate the pigmentation pattern of fish and sea urchins as well as the plate diagrams of caerate crinoids. This generic model is the subject of the rest of this paper and is introduced below [1, 2, 8, 10].

3.2 A General Turing Model

In 1999, Barrio et al. [1] proposed a general Turing system that met the conditions stated in Equations (2.12 - 2.15). While this model was not based on specific chemical

reactions, it has proved useful for the study of different aspects of pattern formation. Barrio et al. have looked at boundary conditions, domain shape, non-linearities, and the coupling of systems [1]. Varea et al. have also explored this model on a sphere [8] and Barrio et al. studied its symmetry breaking properties in a circular domain [2]. Leppänen named this general Turing model the BVAM model in his Doctoral dissertation in which he studied this system in depth [11].

The BVAM model was created by taking generic reaction functions, f and g , and expanding them in a Taylor series about the homogeneous stationary state, (U_0, V_0) , up to the third-order term.

$$\begin{aligned}
 f(U, V) &\approx f(U_0, V_0) + f_U(U_0, V_0)(U - U_0) + f_V(U_0, V_0)(V - V_0) \\
 &+ \frac{1}{2}(f_{UU}(U_0, V_0)(U - U_0)^2 + 2f_{UV}(U_0, V_0)(U - U_0)(V - V_0) + f_{VV}(U_0, V_0)(V - V_0)^2) \\
 &+ \frac{1}{6}(f_{UUU}(U_0, V_0)(U - U_0)^3 + 3f_{UUV}(U_0, V_0)(U - U_0)^2(V - V_0) \\
 &+ 3f_{UVV}(U_0, V_0)(U - U_0)(V - V_0)^2 + f_{VVV}(U_0, V_0)(V - V_0)^3)
 \end{aligned}$$

The expansion for g is the same. Not all of the terms from the expansion are useful in the model. The typical reaction kinetics between the activator and inhibitor are expressed in the uv term and the uv^2 term, which express the effect of the inhibitor on the activator. The u^2v term expresses the reverse-reaction kinetics and is not used in the model. Replacing $u = (U - U_0)$ and $v = (V - V_0)$, we will assume from now on that the partial derivatives are all evaluated at the stationary state, (U_0, V_0) . The functions f and g become

$$\begin{aligned}
f(U, V) &\approx f_U u + f_V v - f_{UV} uv - f_{UVV} uv^2 \\
g(U, V) &\approx g_U u + g_V v + g_{UV} uv + g_{UVV} uv^2.
\end{aligned}
\tag{3.1}$$

The function f describes the reaction kinetics for the activator and g for the inhibitor. Each of the coefficients is represented by a parameter value and the BVAM model is now stated:

$$u_t = D\delta\nabla^2 u + \alpha u + v - r_2 uv - \alpha r_1 uv^2 \tag{3.2}$$

$$v_t = \delta\nabla^2 v + \gamma v + \beta v + r_2 uv + \alpha r_1 uv^2. \tag{3.3}$$

This may also be written

$$\begin{aligned}
u_t &= D\delta\nabla^2 u + \alpha u(1 - r_1 v^2) + v(1 - r_2 u) \\
v_t &= \delta\nabla^2 v + \beta v(1 + \frac{\alpha r_1}{\beta} uv) + u(\gamma + r_2 v).
\end{aligned}
\tag{3.4}$$

as done in Barrio et al. [1]. In this system, $D = D_U/D_V$ is the ratio between the diffusion coefficients. We must have $D < 1$ since the diffusion of the inhibitor must be significantly greater than the activator in order for diffusion-driven instability to occur. The value of δ acts to scale the diffusion by speeding it up or slowing it down. The faster the diffusion, the greater the amount of patterning for the same domain size. In essence, varying δ effectively varies the viewing area.

The coefficients, α , β , and γ describe changes in the reaction rates with respect to the concentrations of u and v (i.e., the partial derivatives from the Taylor expansion). The parameters r_1 and r_2 affect the interaction of the activator and inhibitor and ultimately determine if the pattern at a final steady-state will be spots or stripes. When r_1 and r_2 are both non-zero, the cubic and quadratic terms are present, and

the pattern will result in spots. The absence of quadratic terms will lead to stripes [1].

This general Turing model has two solutions at the uniform stationary state where $f = g = 0$. Rearranging the equation below

$$\alpha u(1 - r_1 v^2) + v(1 - r_2 u) = \beta v(1 + \frac{\alpha r_1}{\beta} uv) + u(\gamma + r_2 v) = 0 \quad (3.5)$$

shows that one solution is $u = v = 0$ and the other is $\frac{-(\alpha+\gamma)}{\beta+1}u = v$. We keep the investigation limited to only one homogeneous steady-state solution at $(0, 0)$ by letting $\gamma = -\alpha$. Based on the linear stability analysis from Section 2.2, our A matrix becomes:

$$A = \begin{bmatrix} \alpha & 1 \\ \gamma & \beta \end{bmatrix}.$$

Therefore, the four conditions for diffusion-driven instability are stated here for the BVAM model.

$$\alpha + \beta < 0 \quad (3.6)$$

$$\alpha(\beta + 1) > 0 \quad (3.7)$$

$$\alpha + \beta D > 0 \quad (3.8)$$

$$\alpha + \beta D > 2\sqrt{\alpha D(\beta + 1)} \quad (3.9)$$

3.3 Parameter Selection

It is our goal to explore the pattern created by stripes from the BVAM on a square, a disk, and the surface of a hemisphere. The parameters used by Calhoun and Helzel [3] will be our starting point. For all our experiments, the boundary conditions

are Neumann, zero-flux, which has the effect of isolating the system from outside influences. We let $\alpha = 0.899$, $\beta = -0.91$, and $D = 0.516$ for all our tests. It can be confirmed that these parameters meet the above conditions for diffusion-driven instability. We let δ vary in our experiments. Since we are only interested in stripes, we let $r_1 = 3.5$ and $r_2 = 0$. We begin our simulations in one spatial dimension and then expand it to two dimensions.

CHAPTER 4

NUMERICAL SIMULATIONS

4.1 The Dispersion Relation and a Pattern in 1-D

In Section 2.2, we identified the conditions for which a diffusion-driven instability would take place in a Turing type reaction-diffusion system. It turns out that the dispersion relation, Equation (2.6), is a key element in this understanding. This relationship between λ and k^2 reveals the range of wavenumbers leading to a positive growth rate. Because we solve the reaction-diffusion system on a discrete domain, the range of wavenumbers is finite. The wavenumber that gives the largest possible positive λ becomes the dominant wavenumber. Thus, it is called the *critical* wavenumber. Figure 4.1 shows the dispersion relation between $Re(\lambda)$ and k for the example shown in Figure 2.3 from Section 2.1 run with parameters from Section 3.3 and $\delta = 0.0021$. We can see in the graph of the dispersion relation that the range of wavenumbers that give a positive λ is approximately [11,16]. It can also be seen that the critical wavenumber is approximately 14.

The value of k can also be determined by using Equation (2.11), which came from finding the minimum value of the determinant of A_k , the point at which instability occurs. We restate this equation applying the parameters from the BVAM model,

$$k^2 = \frac{1}{\delta} \left(\frac{\alpha + \beta D}{2D} \right). \quad (4.1)$$

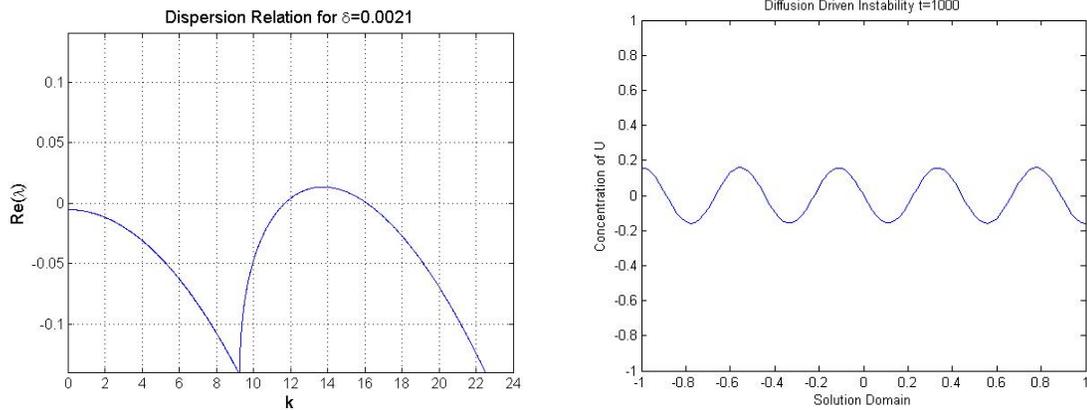


Figure 4.1: The dispersion relation and Turing patterns in one dimension

To illustrate the importance of this equation, we substitute the parameters in Section 3.3 and find that $k \approx 14$. We can use Equation (4.1) then to predict the critical wavenumber for a particular set of parameters.

The wavenumber has a reciprocal relationship to the wavelength, ω . If we look at our particular solution, Equation (2.5), from Section 2.2 for $k = 14$,

$$\mathbf{w}_{14} = \mathbf{C}_{14} e^{\lambda_{14} t} e^{i(14)x} \quad (4.2)$$

we can expand this as

$$\mathbf{w}_{14} = \mathbf{C}_{14} e^{\lambda_{14} t} [\cos(14x) + i \sin(14x)]. \quad (4.3)$$

Because of the format we have chosen for our solution and the fact that sine and cosine have a period of 2π , the specific reciprocal relationship between wavenumber and wavelength is $\omega = 2\pi/k$. It follows for k calculated from Equation (4.1) that $\omega \approx 0.44635$. We can determine the number of waves, N , which should result in our given domain length, L . Since the domain is $[-1, 1]$, $L = 2$. It follows that

$N = L/\omega = 2/0.44635 \approx 4.5$. We can see from the diagram on the right in Figure 4.1 that the pattern formed has the determined 4.5 waves. From this understanding, we turn to two-dimensional domains beginning with the square domain.

4.2 Square Domain

In this thesis, it is our goal to understand why perfect stripes can develop in a square or hemisphere domain but not in a disk. At this point, we have laid out the important elements for a solution in one dimension. We now apply this understanding to two dimensions bounded by a square. The first question that might arise is if our equation defining the critical wavenumber will apply in two dimensions. Looking back at Section 2.2, we can see that the solution, Equation (2.5), in two spatial dimensions would be

$$\mathbf{w}_{\mathbf{k}} = \mathbf{C}_{\mathbf{k}} e^{\lambda_{\mathbf{k}} t} e^{i\mathbf{k} \cdot \mathbf{x}}$$

where \mathbf{k} now represents a vector of two wavenumbers, $(\mathbf{k}_1, \mathbf{k}_2)$. This will result in the same characteristic equation, Equation (2.6), where k^2 is replaced by $\|\mathbf{k}\|^2 = \mathbf{k} \cdot \mathbf{k}$. As a result, our critical wavenumber equation, Equation (4.1), will remain the same. However, we will look at the dominant wavenumber to see if it is always the same in each spatial dimension.

In all the numerical simulations run in one dimension and the two-dimensional square grid, we used a MATLAB code written specifically for these experiments. These codes are available in Appendix A. The initial conditions were always a randomly generated set of values between $[-0.5, 0.5]$. For some experiments, we seeded the random number generator using a prescribed seed for each test and at other times we didn't initialize the random seed generator. After many experiments, it was

determined that seeding or not seeding did not change the overall results.

In all the tests, we maintain a zero-flux boundary condition. This has the effect of isolating the solution from outside influences. We used the Forward Euler method to evolve the solution in time and we used a second-order central finite difference approximation to evaluate the diffusion term. In one dimension, we used 100 grid points, and in two dimensions it was always a 50×50 grid. Since we are interested in the final steady-state, each test was run for 25000 time steps where $\Delta t = 0.04$ in one dimension and at least 37,500 time steps in two dimensions with $\Delta t = 0.08$. In all cases, the number of time steps chosen was enough to see no discernible change in the solution.

As shown in the last section, we can pick the number of waves we want to appear in our grid and from this calculate the proper δ using Equation (4.1). For example, $\omega = L/N$ and $k = 2\pi/\omega$, so $k = 2\pi N/L$. Since the length of the domain in the square grid is 2, then $k = \pi N$. Therefore, if we want 5 waves, we let $k = 5\pi$ and calculate δ by rearranging Equation (4.1) to be

$$\delta = \frac{1}{k^2} \left(\frac{\alpha + \beta D}{2D} \right).$$

In Figure 4.2, we show two simulations where the one on the left is run for 5 waves and the one on the right is run for 5.25 waves. The wave seen in the one-dimensional case in Figure 4.1 is the variation in levels of concentration of the activator, U . In the two-dimensional picture, the change in concentration is represented by colors where red is high concentration and blue is low concentration. The red stripes, then, show the peaks of the waves while the blue stripes show the valleys of the waves. In the picture on the left, we can count the waves by counting the number of red stripes. We



Figure 4.2: The two dimensional BVAM model in a square domain with 5 waves (stripes) on the left and 5.25 waves (stripes) on the right.

see a red stripe along the left boundary so we know the wave starts at its maximum. To count the number of waves, we begin counting from the boundary with the first wave being the first stripe completely inside the square. In this way, we see that there are 5 waves (or stripes) in the horizontal direction across the square. The picture on the right, which was run for 5.25 waves (or stripes), shows 5 vertical stripes across the top of the square, but it also shows horizontal stripes near the bottom of the grid. There are clearly three stripes formed by waves in the vertical direction, with some form of undulation showing in the stripes from the horizontal direction. Because we impose a zero-flux on the boundary, the wave is forced to have a peak or valley at the boundary. If the wavenumber that is imposed does not allow for an integer or half-integer number of waves, we expect to see defects of the sort shown in Figure 4.2 on the right. As a result, in all our testing, we only choose integer or half-integer number of waves to test.

We ran tests for consecutive integer and half-integer values from 3 stripes to 6.5 stripes each with the same random seed. Table B.1 in Appendix B records the k value and δ value for each test. In Figure 4.3, we show two results. Typically, the waves aligned, giving either all vertical or all horizontal stripes. We can see in Figure 4.3 the

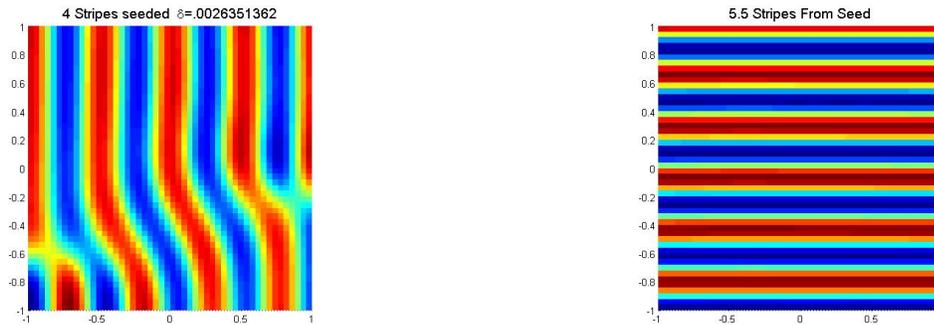


Figure 4.3: The result of integer number of stripes on the left and half-integer number of stripes on the right.

difference between an integer number of stripes and a half-integer number of stripes. If the stripes are either red on both sides or blue on both the sides parallel to the stripes, this is an integer number of stripes. If the stripes begin with red and end with blue, this would be an example of a half-integer number of stripes.

Sometimes the waves would not end up fully aligned in one direction in which case we can see stripes forming in each direction. Figure 4.4 shows two examples of this. Both tests were run with the unseeded random initial conditions. We can see

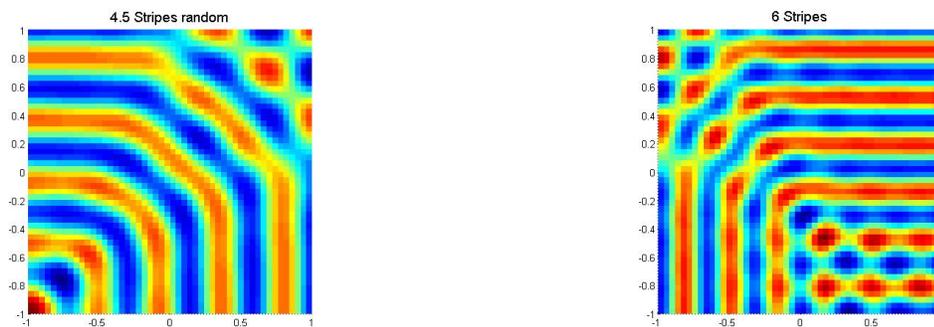


Figure 4.4: Predicted number of waves in each direction: 4.5 stripes on the left and 6 stripes on the right.

that we get 4.5 stripes in each direction for the figure on the left and 6 stripes in each direction for the figure on the right.

While the process of getting a predicted number of stripes was robust in at least one direction, there were times when the competition between waves in the different directions produced mixed results. In Figure 4.5, we show two examples of this. We

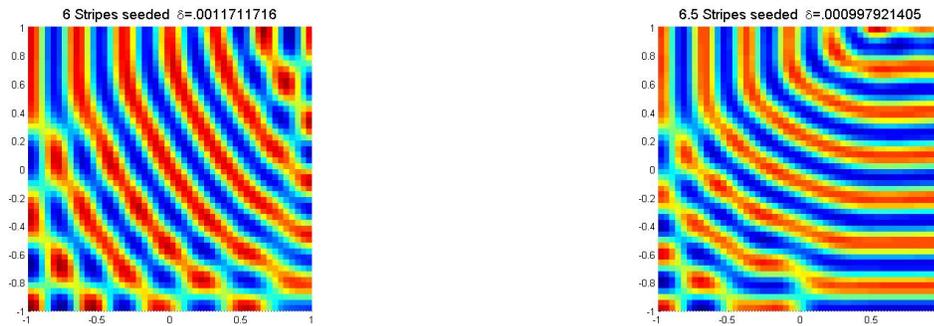


Figure 4.5: Competition between horizontal waves and vertical waves can cause defects.

can count 6 stripes across the top of the left-hand figure while the one on the right has 6.5 stripes down the right side of the square, but a count of the stripes on the other boundaries does not give the predicted amount.

In the last set of examples for the square grid, we show that straight stripes with no defects (i.e., perfect) can be formed for various numbers of stripes and in different directions. Figure 4.6 shows perfect stripes from random initial conditions for both 3.5 stripes and 5.5 stripes.

We have shown that perfect stripes can be formed in a square grid. We have also confirmed that the number of stripes can be chosen for a two-dimensional solution based on the length of the domain and the critical wavenumber equation; see (4.1). We chose values for k that would give us waves that would fit exactly into the domain length, keeping the zero-flux boundary conditions from causing avoidable defects. We now apply this understanding to the two-dimensional circular domain.



Figure 4.6: Perfect Stripes

4.3 Circular Domain

4.3.1 Analysis of the BVAM Model on a Disk

We begin by exploring how the solution to the BVAM Turing system is affected by the shape of the domain. While a square domain lends itself to solutions in rectangular coordinates, a circular domain lends itself to polar coordinates. The two spatial dimensions are the radial dimension r , and the angular dimension θ . Our analysis here follows the analysis laid out by Aragón et al. [10]. In what follows, we assume our disk has radius one.

The solution on a disk applies a Fourier series in the angular dimension and a Bessel series in the radial dimension. The form of the solution for a particular wavenumber, k is

$$\mathbf{w}_k = \mathbf{C}_m e^{\lambda_k t} J_m(kr) e^{im\theta} \quad (4.4)$$

where $\mathbf{C}_m \in \mathbb{R}^{2 \times 1}$ are the Fourier coefficients and J_m are the Bessel functions of the first kind of order m where m is an integer mode. The solutions $\mathbf{w} = [u, v]^T$ are $u = u(r, \theta, t)$ and $v = v(r, \theta, t)$. Substituting this solution into Equation (2.4), we get the same characteristic equation as we did in Section 2.2 and Equation (2.6). Aragón

et al. [10] show that for a normalized k and a unit disk, the dispersion relation leads to the same critical wavelength. We again state this critical wavelength, k_c , at which at least one λ becomes positive, producing the necessary diffusion-driven instability:

$$k_c^2 = \frac{1}{\delta} \left(\frac{\alpha + \beta D}{2D} \right). \quad (4.5)$$

While k can be calculated from Equation (4.5), the mode m cannot be determined this way. We must decompose the solution to find the values of m .

The solutions u and v at a final steady-state under zero-flux boundary conditions can be expanded in a Fourier-Bessel series. For example:

$$u(r, \theta) = \sum_{m=0}^{\infty} \sum_{n=1}^{\infty} C_{mn} J_m(k_{mn}r) e^{im\theta} \quad (4.6)$$

where $k_{mn} = \kappa_{mn}/a$, κ_{mn} is the n th zero of the derivative of the Bessel function J'_m , and a is the radius of the disk. The solution can be decomposed to reveal the principal modes by computing the magnitude of the complex coefficient, C_{mn} [10].

$$C_{mn} = \frac{k_{mn}^2}{\pi[(k_{mn}a)^2 - m^2][J_m(k_{mn}a)]^2} \int_0^a \int_0^{2\pi} u(r, \theta) r J_m(k_{mn}r) e^{-im\theta} d\theta dr. \quad (4.7)$$

From the decomposition, we can reconstruct the solution to show that the decomposition retains the same key features as the original solution.

In Figure 4.7, we show a solution and its reconstructed form, which is a truncated series solution, using coefficients C_{mn} computed from a numerical solution, as described in Section 4.3.2 below. The figure on the left is the numerical solution and the one on the right is the reconstructed figure. The key features are the areas of high concentration and low concentration. The original solution has darker reds and

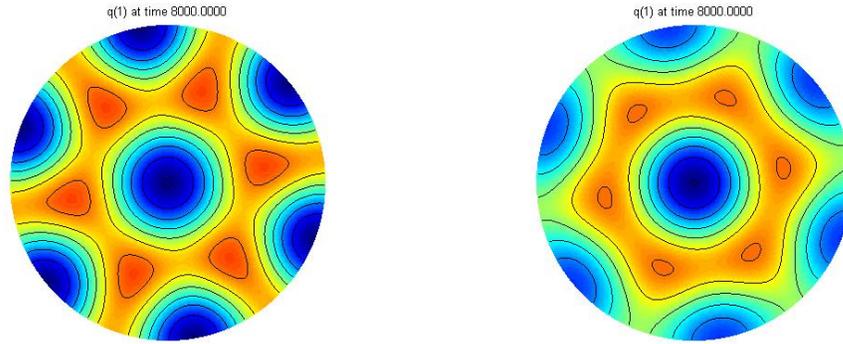


Figure 4.7: Solution on a disk and its reconstructed form from Fourier-Bessel coefficients where $q(1)$ corresponds to activator species U .

blues, indicating the levels of concentrations in the reconstructed figure are not as high or as low. However, the peaks and valleys are in the same place, showing the same pattern in the reconstructed result as in the original. It should also be noted here, that this particular solution reveals the type of waves generated in a circular domain. We can see the rings from the center out indicating waves in the radial direction. We can also see the spoke-like formation formed by waves in the angular direction. This solution does not reveal the stripes we are looking for, but it is an example of some of the unusual patterns that we came across. As we go on to look at stripes formed on the disk, this picture is useful for understanding the nature of the solutions on the circular domain.

4.3.2 Numerical Method and Results

The solution to the BVAM model on a disk was solved using a Runge-Kutta Chebyshev method for the time step and the finite volume method developed by Calhoun and Helzel [3] solved over a mesh grid of 100×100 . The boundary conditions are zero-flux given by,

$$\mathbf{n} \cdot \nabla u(r, \theta) = \mathbf{n} \cdot \nabla v(r, \theta) = 0$$

for all $\theta \in [0, 2\pi)$ and r at the boundary with \mathbf{n} the unit outward normal. For each test, $\Delta t = 1$ and 10,000 time steps were performed.

To approximate the integral needed to compute the Fourier-Bessel coefficients, we use the fact that the numerical solution to our finite volume scheme is a *cell averaged* value. The discrete solution satisfies

$$u_{ij}^n \approx \frac{1}{\text{Area}(C_{ij})} \int_{C_{ij}} q(x, y, t_n) dA \quad (4.8)$$

where C_{ij} is mesh cell ij , $\text{Area}(C_{ij})$ is the area of mesh cell C_{ij} , and $q(x, y, t_n)$ is the exact solution at time level t_n . This approximation is second order for a second-order scheme. To approximate our Fourier-Bessel coefficients, we can replace the integral in Equation(4.7) with a discrete sum over an $M \times N$ mesh, using approximate mesh cell areas as weights for the sum. We have then

$$C_{mn} \approx \frac{k_{mn}^2}{\pi[(k_{mn}^2 - m^2][J_m(k_{mn})]^2)} \sum_{i=1}^M \sum_{j=1}^N u_{ij} J_m(k_{mn} r_{ij}) e^{(-im\theta_{ij})} dA_{ij} \quad (4.9)$$

where the area element $dA_{ij} \approx r dr d\theta$, and (r_{ij}, θ_{ij}) are computed by converting the Cartesian components of the mesh cell centers to polar coordinates.

The stripes on the disk tend to align in one direction. As with the square domain, we can calculate the number of stripes, N_r , that go across the diameter. The domain length, L_r , is twice the radius, so we have $k/\pi = N_r$. From this, δ can be determined by Equation (4.5). The modal decomposition gives m , which from Equation (4.6) can be seen to be the wavenumber in the angular direction. We take the length of the domain, L_θ , in this direction to be the circumference so that $L_\theta = 2\pi$. Therefore,

the relationship between m and the number of waves, N_θ , around the boundary is $m = N_\theta$. In all the pictures of the circular figures, the time at steady-state is listed above the circle. Table B.2 gives the number of stripes across the diameter and the number of waves around the circumference based on choosing k as an integer from 10 to 18. We do not show all of the pictures here, but select ones in this section for discussion. They are all shown together in Section B.2, Appendix B.

Since the waves generated in a circular domain occur in the angular and radial directions, we are interested in the waves around the boundary compared to the waves (stripes) across the diameter. In Figure 4.8, we compare the stripes across the diameter to the waves around the circumference and look at the values for k and m . We count the stripes across by counting along a diameter perpendicular to the

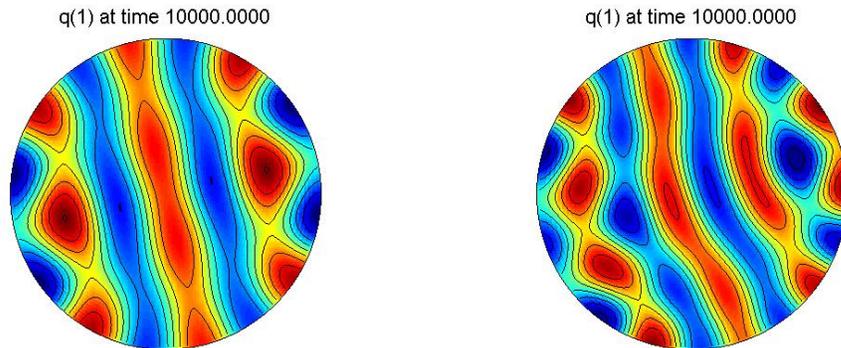


Figure 4.8: Stripes on a disk for $k = 10$, $m = 8$ on the left and $k = 12$, $m = 10$ on the right

stripes. The figure on the left has 3 stripes and the one on the right has almost 4 full stripes. To calculate the number of stripes across, we use $N_r = k/\pi$. In this way, we calculate 3.1 stripes when $k = 10$ and 3.8 when $k = 12$. To count the waves around the boundary, we look at all the high points where red either touches the boundary or is close enough to cause a high spot on the rim. We can see the picture on the

left has 8 waves around the perimeter and the one on the right has 10 waves at the edge. In Figure 4.9, we show the chart of decomposed coefficients up to $m = 40$ and $n = 39$ for $k = 10$ and $k = 12$. Since our grid size was 100×100 , we calculated more coefficients than are shown here, choosing to cut off the lowest modes which are not relevant to this discussion. We present all of the highest modes in the diagram. The

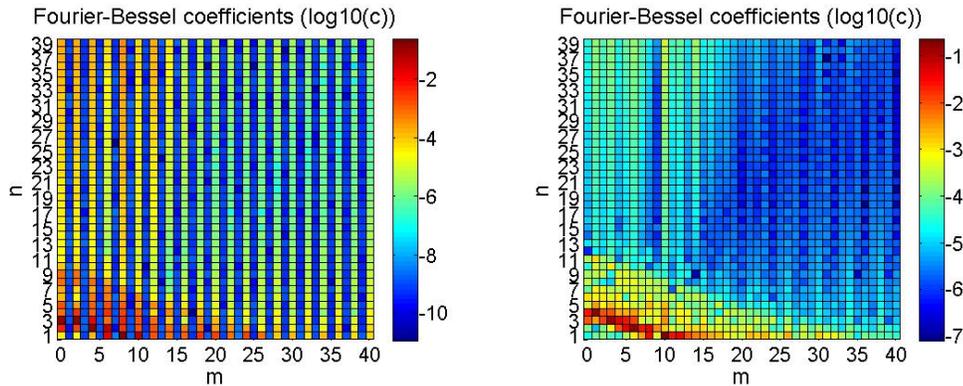


Figure 4.9: Fourier-Bessel coefficients for $k = 10$ on the left and $k = 12$ on the right darker the red, the larger the value of the coefficient. In Figure 4.9, the largest mode in the left-hand figure is the darkest red square at $m, n = 8, 1$. The right-hand figure shows the darkest red square at $10, 1$. We note here that the number of waves on the boundary for each figure in Figure 4.8 agrees with the value of m for the largest Fourier-Bessel coefficient from Figure 4.9

It is not easy to tell the largest coefficient from the above charts. It is evident that there are many modes that are playing a role in the solution on a disk. As k gets larger, the number of modes that are somewhat dominant increases so that the primary mode, m , is not always equal to the number of waves on the perimeter. In Figure 4.10, we show the solution for $k = 17$ and the corresponding Fourier-Bessel coefficient chart. We also show the actual numerical values in Table B.3, Appendix B. There are 5 stripes across and 10 clear high points or waves around the boundary.

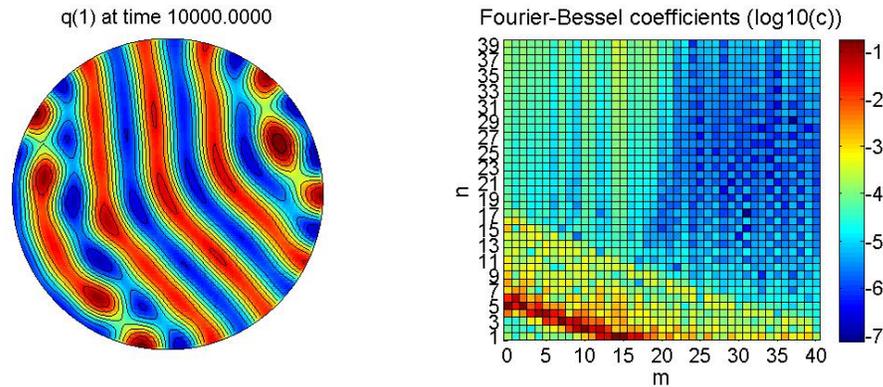


Figure 4.10: The solution and the corresponding Fourier-Bessel coefficients for $k = 17$

Table B.3, Appendix B shows 6 modes all within 0.02 of each other in size. We can see, $k_{0,5}$, $k_{2,5}$, $k_{7,3}$, $k_{10,2}$, $k_{12,1}$, and $k_{15,1}$ are the modes that have the largest effect on the solution with $k_{0,5}$ and $k_{15,1}$ being the two dominant modes. Since the number of waves on the boundary is 10, it is clear the largest modes are not significantly large enough to force the number of waves on the boundary to be 15.

Looking at Table B.2, we can see a pattern developing between N_r and N_θ . As k increases, the number of stripes across and waves around also increases at a similar rate. The stripes in the middle are less wavy and more straight. We can see from Figure 4.11 below just how nice the stripes look for $k = 18$. But even here, the stripes on the ends remain very wavy. From these tests, we can see that given a k value, the number of stripes predicted is accurate. For most of the tests, the dominant m mode was also the same as the waves on the boundary. We now look to the hemisphere and the patterns formed on it.

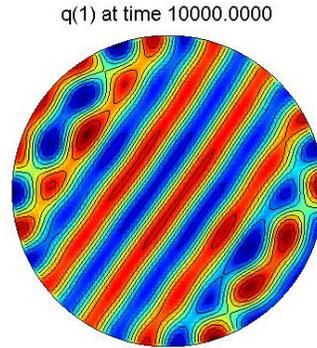


Figure 4.11: Solution for $k = 18$

4.4 Hemisphere

For both the square and the disk, we focused on the number of waves in each spatial dimension. As we apply the BVAM model to the hemisphere, we want to continue focusing on the number of waves in each spatial dimension. For the hemisphere, the two spatial dimensions are around the boundary and over the top of the hemisphere. We use the RKC method combined with a finite volume scheme and the code generated by Calhoun and Helzel [3]. The solution is solved on a 100×100 grid with $\Delta t = 1$ and 10,000 time steps. We again apply Neumann boundary conditions and use the same random initial conditions. We use most of the same values for k and δ as we did for the disk. Table B.4 in Appendix B shows the results for the number of stripes over the top and the number of waves on the boundary. We show all the figures in Appendix B and select a few here to discuss.

Figure 4.12 shows the solutions for $k = 8$ and $k = 11$. To count the stripes over the top, count each full red stripe. If there is a red circle at the end, this counts as half a stripe. To count the waves around simply count each time the red touches the boundary. The figure on the left has 3.5 stripes over the top and 7 waves around the

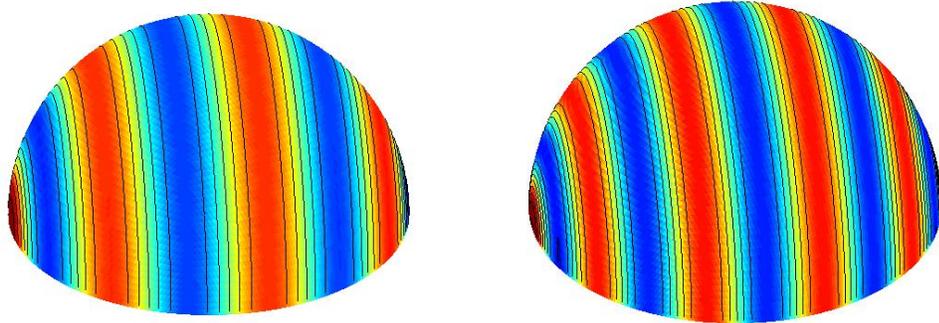


Figure 4.12: Hemisphere with $k = 8$ and $k = 11$

boundary. The figure on the right has 5 stripes over the top and 10 waves around the boundary. The stripes are perfect. In fact, the stripes are perfect for each test up through $k = 14$. The last test revealed a slight defect for $k = 15$. Figure 4.13 shows $k = 15$ from both sides. The solution to the BVAM model on a hemisphere robustly

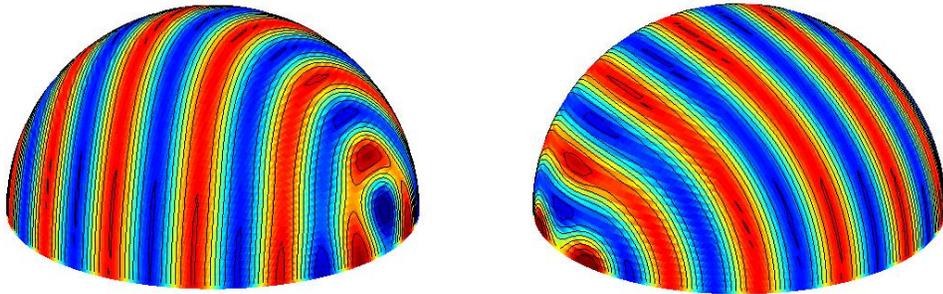


Figure 4.13: Hemisphere with slight defect

produces stripes. Each stripe meets the boundary in two places, thus the number of waves around is always twice as many as the number of waves over the top.

CHAPTER 5

CONCLUSIONS

5.1 Perfect Stripes in Different Geometries

We have shown that perfect stripes can be formed on a square and hemisphere. We have shown good stripes on a disk can be obtained, but only in the middle of the domain. We have shown that the number of stripes on a square and disk can be predicted by connecting the critical wavenumber equation to the length of the domain in each spatial dimension. While each geometry produces some defects, producing stripes in general is dependable in all domains. And, in all geometries, we can produce a chosen number of stripes. But we are most interested in determining if it is possible to produce perfect (i.e., straight with no defect) stripes on a disk from this general Turing model, starting from random initial conditions and applying zero-flux boundary conditions.

Table 5.1 shows a comparison of the lengths, L , of the domains in each spatial dimension for each type of geometry. It also compares the number, N , of waves in one spatial dimension with the number of waves in the other spatial dimension. The length of the domain in the square is 2 in each dimension. In general, there were two main results from the solution on a square. Either the waves aligned in predominantly one direction or they filled half with horizontal stripes and half with vertical stripes. In general, when the waves did not align in one direction, the number of stripes in

Table 5.1: Modal comparison by geometry

Geometry	$L_1 : L_2$	$N_1 : N_2$
Square	1 : 1	1 : 1
Disk	1 : π	1 : 2.6
Hemisphere	1 : 2	1 : 2

each spatial dimension was the same. Thus, the ratios from Table 5.1 for the square show that the ratios for each are 1 : 1.

The spatial domains for the hemisphere were over the top (the polar direction) and around the boundary. The distance over the top is π and the distance around the boundary is 2π . The ratio between the lengths of the domains then is 1 : 2. It was also shown in Section 4.4 that the number of stripes around was always twice the number over the top. Again, the ratio is 1 : 2.

The length of the domain in the radial direction for the disk is twice the radius or 2. The length of the domain in the angular direction is the circumference of the boundary or 2π . The ratio of the lengths then is 1 : π . Below in Table 5.2, we show again values from Table B.2, adding a column with the ratio between the waves around and the waves across. This shows an approximate ratio between the number of waves in each spatial dimension on the disk to be 1 : 2.6.

If the ratio between the lengths of the domains is the same as the ratio between the number of waves in each dimension, it is easy to see the waves should fit neatly into the given lengths. Thus, for the square and hemisphere, we can get perfect stripes. The ratios for the disk simply do not match up. The waves across the circle do not fit neatly with the waves around the circle. The waves on the boundary keep the stripes near the boundary from being perfect. Our numerical results strongly indicate that

Table 5.2: Ratio between stripes across the disk and waves around the disk

Stripes across N_r	Waves around N_θ	N_θ/N_r
3.1	8	2.6
3.5	9	2.6
3.8	10	2.6
4.1	11	2.7
4.5	12	2.7
4.8	13	2.7
5.1	13	2.5
5.4	14	2.6
5.7	15	2.6

the solution to this general Turing model with zero-flux boundary conditions on a disk will not generate perfect stripes.

5.2 Exploring Stripes on Other Domains

There is still more to explore regarding stripes in a circular domain. We need to study the Fourier-Bessel solutions in greater detail, attempting to understand the connection between k and m . We also need to explore different boundary conditions and expand to other domain shapes.

When Turing first proposed the possibility of the existence of morphogens, it was to propose a possible explanation for pattern development in living organisms. As such, if one is to make a contribution toward this end, then it would be important to explore this Turing model further in domains that are more similar to living organisms. While the disk is the most relevant domain we studied, it would be good to also look at the surface of a sphere and a torus. If we wanted to make any sort of connection between stripes on the hide of a mammal, we would need an irregular type of domain

that resembles the shape of the animal.

J. D. Murray wrote the textbook, “Mathematical Biology” [5], which deals extensively with the mathematical understanding needed to solve reaction-diffusion equations. But even this understanding is often limited to the simplest mechanisms and often in one spatial dimension because of the complexity of these models. There is more to do in gaining a greater understanding of complex reaction-diffusion equations and their solutions. We look forward to future developments in this area.

REFERENCES

- [1] R. Barrio, C. Varea, J. Aragón, and P. Maini. “A two-dimensional numerical study of spatial pattern formation in interacting Turing systems”. *Bull. Math. Biol.*, vol. 61 (1999), pp. 483-505.
- [2] R. A. Barrio, J. L. Aragón, M. Torres, and P. K. Maini, (2002). Size-dependent symmetry-breaking in models for morphogenesis. *Physica D*, 168-169:61.
- [3] D. Calhoun, C. Helzel, “A finite-volume method for solving parabolic equations on logically Cartesian curved surface meshes”, *SIAM J. Sci. Comp.*, Vol. 31, Issue 6. pp. 4066-4099 (2009).
- [4] P.K. Maini, K.J. Painter, H.N.P. Chau, “Spatial pattern formation in chemical and biological systems”, *Faraday Transactions*, 93 (20), 3601-3610 (cover article) (1997)
- [5] J. D. Murray. *Mathematical Biology*. 2nd Edition, Springer Verlag, Berlin, 1993.
- [6] Allen R Sanderson, Mike Kirby, Chris R Johnson, and Lingfa Yang. ”Advanced Reaction-Diffusion Models for Texture Synthesis,” In *Journal of Graphics Tools* Vol. 11, No. 3: p47-71, 2006.
- [7] A. M. Turing. “The chemical basis of Morphogenesis.” *Phil. Trans. R. Soc. Lond. B* 1952 237, pp. 37–72.
- [8] C. Varea, J. L. Aragón, and R. A. Barrio. “Turing patterns on a sphere”. *Physical Review E*, vol. 60(4), (1999), pp. 4588–4592.
- [9] M. A. J. Chaplain, M. Ganesh, I. G. Graham. “Spatio-temporal pattern formation on spherical surfaces: numerical simulation and application to solid tumour growth”, *J. Math. Biol.*, 42, 387-423 (2001).
- [10] J. Aragón, M. Torres, D. Gil, R. A. Barrio, P. K. Maini. “Turing patterns with pentagonal symmetry”, *Physical Review E*, Vol. 65, 051913, (2002).
- [11] Teemu Leppänen. “Computational Studies of Pattern Formation in Turing systems”, Doctoral dissertation, Helsinki University of Technology, 2004. ISBN 951-22-7395-0.

APPENDIX A

MATLAB CODE

A.1 One-Dimension Code for Square Domain

```

%Solve a Turing model system of equations in 1-D space over time. Apply
%Euler's Method to a semi-discretized Reaction-Diffusion system.

%clear all

%Grid size
Tf=3000;
a=-1;           % Lower boundary
b=1;           % Upper boundary
M=100;         % M is the number of spaces between points a and b.
dx=(b-a)/M;    % dx is delta x
x=linspace(a,b,M+1); % M+1 equally spaced x vectors including a and b.

%Time stepping
dt=0.04;       %100*(dx^2)/2; % dt is delta t the time step
N=Tf/dt;      % N is the number of time steps in the interval [-1,1]

%Constant Values
D=0.516;       % D is the Diffusion coefficient Du/Dv
delta=0.0021;  % sizes the domain for particular wavelengths
alpha=0.899;   % a is alpha, a coefficient in f and g (-a is gamma)
beta=-0.91;    % b is beta, another coefficient in f and g
r1=3.5;        % r1 is the cubic term
r2=0;          % r2 is the quadratic term
gamma=-alpha;  % g is for gamma

%pre-allocation
unp1=zeros(M+3,1);

```

```

vnp1=zeros(M+3,1);

%Initial Conditions
un=-0.5+rand(M+3,1);    %Begin with a random point between [-0.5,0.5]
vn=-0.5+rand(M+3,1);

for n=1:N
    un(1)=un(3);        %Boundary conditions on left flux is zero
    un(M+3)=un(M+1);    %Boundary conditions on right
    vn(1)=vn(3);
    vn(M+3)=vn(M+1);

for j=2:M+2
    %Source function for u and v
    srcu=alpha*un(j)*(1-r1*vn(j)^2)+vn(j)*(1-r2*un(j));
    srcv=beta*vn(j)*(1+(alpha*r1/beta)*un(j)*vn(j))+un(j)*(gamma+r2*vn(j));

    Lapu=(un(j-1)-2*un(j)+un(j+1))/dx^2;        %Laplacian u
    Lapv=(vn(j-1)-2*vn(j)+vn(j+1))/dx^2;        %Laplacian v

    unp1(j)=un(j)+dt*(D*delta*Lapu+srcu);
    vnp1(j)=vn(j)+dt*(delta*Lapv+srcv);
end
un=unp1;
vn=vnp1;

% Graphing
if mod(n,25000)==0

    %subplot(2,1,2)
    plot(x,un(2:M+2))
    axis([ -1, 1, -1 ,1]);
    fprintf('Time t = %f\n',n*dt);
    input('Hit enter to continue :')
end

end

```

A.2 Two-Dimension Code for Square Domain

```

%Solve a Turing model system of equations in 2-D space over time.  Apply
%Euler's Method to a semi-discretized Reaction-Diffusion system.

%clear all

%Grid size
Tf=1000000;
a=-1;           % Lower boundary
b=1;           % Upper boundary
M=50;         % M is the number of spaces between points a and b.
dx=0.04;      % (b-a)/M; % dx is delta x
dy=0.04;      % (b-a)/M;
x=linspace(a,b,M+1); % M+1 equally spaced x vectors including a and b.
y=linspace(a,b,M+1);

%Time stepping
dt=0.08;      %100*(dx^2)/2; % dt is delta t the time step
N=Tf/dt;     % N is the number of time steps in the interval [0,1]

%Constant Values
D=0.516;     % D is the Diffusion coefficient Du/Dv
delta=0.0021; % sizes the domain for particular wavelengths
alpha=0.899; % a is alpha, a coefficient in f and g (-a is gamma)
beta=-0.91;  % b is beta, another coefficient in f and g
r1=3.5;     % r1 is the cubic term
r2=0;      % r2 is the quadratic term
gamma=-alpha; % g is for gamma

%pre-allocation
unp1=zeros(M+3,M+3);
vnp1=zeros(M+3,M+3);

%Initial Conditions
un=-0.5+rand(M+3,M+3); %Begin with a random point between [-0.5,0.5]
vn=-0.5+rand(M+3,M+3);

for n=1:N

    for i=2:M+2

```

```

        un(i,1)=un(i,3);           %Boundary conditions on left flux is zero
    un(i,M+3)=un(i,M+1);         %Boundary conditions on right
    vn(i,1)=vn(i,3);
    vn(i,M+3)=vn(i,M+1);
    end
    for j=2:M+2
        un(1,j)=un(3,j);         %Boundary conditions on left
        un(M+3,j)=un(M+1,j);     %Boundary conditions on right
        vn(1,j)=vn(3,j);
        vn(M+3,j)=vn(M+1,j);
    end

for i=2:M+2
for j=2:M+2

    %Source function for u and v
    srcu=alpha*un(i,j)*(1-r1*vn(i,j)^2)+vn(i,j)*(1-r2*un(i,j));
    srcv=beta*vn(i,j)*(1+(alpha*r1/beta)*un(i,j)*vn(i,j))+un(i,j)*(gamma+r2*vn(i,j));

    uxx=(un(i-1,j)-2*un(i,j)+un(i+1,j))/dx^2;           %Laplacian u
    vxx=(vn(i-1,j)-2*vn(i,j)+vn(i+1,j))/dx^2;           %Laplacian v
    uyy=(un(i,j-1)-2*un(i,j)+un(i,j+1))/dy^2;           %Laplacian u
    vyy=(vn(i,j-1)-2*vn(i,j)+vn(i,j+1))/dy^2;           %Laplacian v
    Lapu=uxx+uyy;
    Lapv=vxx+vyy;

    unp1(i,j)=un(i,j)+dt*(D*delta*Lapu+srcu);
    vnp1(i,j)=vn(i,j)+dt*(delta*Lapv+srcv);

end
end
    un=unp1;
    vn=vnp1;

% Graphing
if mod(n,6250)==0

    %subplot(2,1,2)
    hdl = surf(x,y,un(2:M+2,2:M+2));
    set(hdl,'edgecolor','none');
    axis([-1, 1,-1,1]);
    %caxis([-10,15]);

```

```
view(2);
colorbar;
fprintf('Time t = %f\n',n*dt);
ch = input('Hit enter to continue :','s');
if (strcmp(ch,'k') == 1)
    keyboard;
end
end
end
```

APPENDIX B

TABLES AND FIGURES FROM NUMERICAL SIMULATIONS

B.1 Square

Table B.1: Values of k and δ based on the number of stripes

Number of stripes	k	δ
3	9.4	0.0046846866
3.5	11.0	0.0034418106
4	12.6	0.0026351362
4.5	14.1	0.0020820829
5	15.7	0.0016864872
5.5	17.2	0.0013937911
6	18.8	0.0011711716
6.5	20.4	0.000997921405

B.2 Disk

Table B.2: Values which correspond to Figure B.1

k	δ	Stripes across N_r	Primary mode- k_{mn}	Waves around N_θ
10	0.0041612403	3.1	$k_{8,1}$	8
11	0.0034390416	3.5	$k_{9,1}$	9
12	0.0028897502	3.8	$k_{10,1}$	10
13	0.0024622724	4.1	$k_{11,1}$	11
14	0.0021230818	4.5	$k_{12,1}$	12
15	0.0018494401	4.8	$k_{13,1}$	13
16	0.0016254845	5.1	$k_{13,1}$	13
17	0.0014398755	5.4	$k_{05,1}$	14
18	0.0012843334	5.7	$k_{15,1}$	15

Figure B.1 shows the results in order from $k = 10$ to $k = 18$ left to right and top to bottom. To count the number of waves around the circumference, count the number of times red touches the outside of the circle.

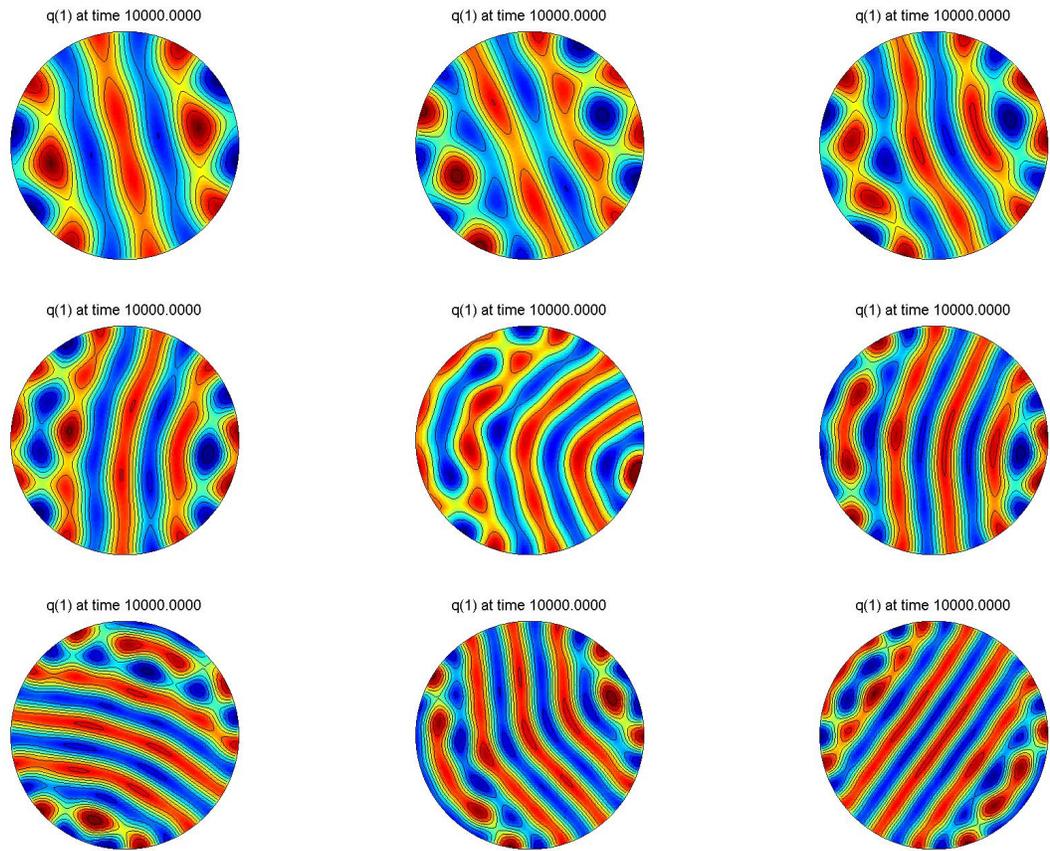


Figure B.1: Results selecting k as an integer from $k = 10$ to $k = 18$

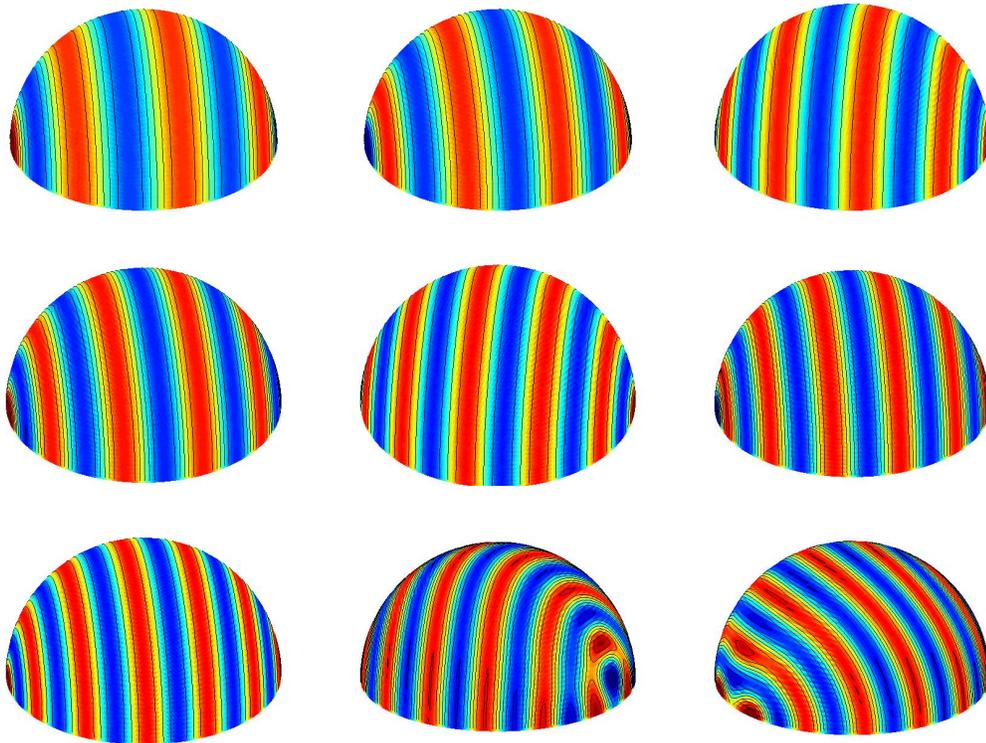
B.3 Hemisphere

Table B.3: Fourier-Bessel coefficients from $k = 17$ in Figure 4.10

k_{mn}	1	2	3	4	5	6	7
0	1.67E-04	6.13E-05	3.06E-05	4.92E-03	1.69E-01	1.52E-02	9.50E-04
1	2.50E-05	3.26E-05	2.77E-05	9.35E-04	2.44E-02	5.95E-02	2.63E-03
2	1.39E-04	3.77E-05	8.67E-05	2.96E-03	1.54E-01	5.24E-03	2.35E-03
3	2.24E-04	1.25E-05	8.14E-04	2.66E-02	6.96E-03	1.66E-03	1.46E-04
4	2.40E-05	9.25E-05	4.87E-04	9.39E-02	1.79E-02	3.62E-03	2.65E-05
5	1.55E-04	3.14E-04	1.49E-02	9.06E-02	2.59E-03	4.51E-04	1.42E-03
6	9.49E-05	4.96E-04	1.09E-02	2.52E-02	2.68E-03	9.84E-05	3.74E-04
7	9.39E-05	1.38E-03	1.54E-01	1.49E-02	2.26E-03	7.81E-05	3.75E-04
8	7.55E-05	1.08E-02	3.44E-02	4.15E-04	3.00E-04	9.14E-04	2.04E-04
9	2.89E-04	6.17E-02	4.31E-02	4.72E-03	6.53E-04	9.63E-04	2.00E-05
10	1.28E-03	1.49E-01	1.06E-02	4.29E-03	1.72E-03	1.06E-04	5.71E-04
11	2.27E-03	3.48E-02	5.18E-03	1.73E-03	1.10E-03	4.48E-04	1.75E-04
12	3.62E-02	6.66E-02	5.49E-03	4.96E-04	1.44E-03	1.53E-03	1.24E-03
13	4.12E-02	1.10E-02	8.75E-04	9.10E-04	3.07E-04	1.05E-04	2.39E-04
14	1.58E-01	1.22E-03	1.64E-03	1.81E-03	1.53E-03	9.74E-04	6.19E-04
15	1.63E-01	1.67E-03	2.62E-03	1.50E-03	1.18E-03	8.88E-04	3.34E-04

Table B.4: Number of stripes on a hemisphere given the k value

k	δ	Stripes over the top N_t	Waves around boundary N_b
8	0.006501938	3.5	7
9	0.0051373337	4	8
10	0.0041612403	4.5	9
11	0.0034390416	5	10
12	0.0028897502	5.5	11
13	0.0024622724	6	12
14	0.0021230818	6.5	13
15	0.0018494401	7	14

Figure B.2: Hemisphere with $k = 8, 9, 10$ in row 1, $k = 11, 12, 13$ in row 2, and $k = 14, 15a, 15b$ in row 3.