

Looking at Patterns: Topological Analysis of Reaction-Diffusion Models

A Thesis
Presented to
The Division of Mathematics and Natural Sciences
Reed College

In Partial Fulfillment
of the Requirements for the Degree
Bachelor of Arts

Joel E. Hawkins

May 2015

Approved for the Division
(Physics)

Daniel Borrero

Acknowledgements

It has only recently dawned on me what this process has meant to me. Post-ironic affection aside, I genuinely love and appreciate everything that this school has done for me; how it has motivated me, inspired me academically and artistically, frustrated me, and pushed me to the end of my rope.

Of course, it is not the institution of Reed itself I have to thank. It is the individuals: my friends, my colleagues, and my mentors. I must first acknowledge my advisor Daniel for holding my hand through this entire exercise, guiding me through darkness when I felt like I was just hiding here. Next, I owe much of my growth as an artist to Marisa who never let my (many) opinions go unchallenged. I could not show enough gratitude towards my mother and father who have encouraged me in every endeavor. I must lastly thank Drew, Theo, and Monty. You have supported me when I've failed, congratulated me in my successes, humored me in failure, and greatly shaped who I claim to be in this moment. Your close friendship has truly been the most meaningful aspect of my adult life; for that, I reserve a special place for the memories of our time at Reed.

Preface

I wrote this thesis with the intention of making it as accessible as possible. However, given the visual nature of pattern formation, it's difficult to capture the most compelling aspects of this project on paper (i.e. the cool pattern videos). For this reason I supplement the standalone thesis presented here with a web page designed to enhance the reader's understanding of the problem. As of writing, the page <http://joelhawkins.info/thesis> provides an overview of this thesis and displays video simulations of each pattern discussed as well as interactive charts of the results derived in Chapter 4.

Table of Contents

Introduction	1
Chapter 1: Reaction-Diffusion Systems	5
1.1 The Gray-Scott model	6
1.2 Numerical simulation	8
Chapter 2: Computational Homology	11
2.1 Homology overview	11
2.1.1 Prior work	14
2.2 Cubical homology	15
Chapter 3: Methods and Procedures	25
3.1 Obtaining Betti numbers	25
3.1.1 Thresholding	26
3.1.2 Computational homology	29
3.2 Calculating entropy	29
Chapter 4: Results	33
4.1 Entropy maps	33
4.2 Transient states	34
4.3 Domain size	36
4.4 Other systems	36
Conclusion	39
Appendix A: Supplemental Math	41
A.1 Extra definitions and theorems	41
Appendix B: Code	43

B.1	Gray-Scott simulation code	43
B.2	Entropy calculation code	45
References		47

List of Figures

1.2 Patterns of chemical concentration U identified in [18]. Each pattern, Figure 1.2a—Figure 1.2l, is designated by a Greek letter which corresponds to the plot in Figure 1.3. Red and blue indicate $U = 1$ and $U \approx 0.2$ respectively. Note that a concentration plot of chemical V would appear as the inverse of U with red and blue swapped. Video simulations of each of these pattern types are available online at http://joelhawkins.info/thesis .	9
1.3 The mapping of Greek letters in Figure 1.2 to their location in k, F parameter space. R and B indicate that the system evolved to uniform red and blue states respectively. This figure also represents a phase diagram of the reaction kinetics. Between the solid and dotted line, the system is bistable for which there are two linearly stable steady states. As f passes below the dotted line, the non-trivial steady state becomes unstable through Hopf bifurcation giving stable periodic orbits for $k < 0.035$ and unstable ones for $k > 0.035$. The trivial state, $(U = 1, V = 0)$ exists for all (f, k) outside the solid line. Adapted from J. E. Pearson, “Complex patterns in a simple system,” <i>Science</i> , vol. 261, no. 5118, pp. 189–192, 1993.	10
2.1 The homology of the Reed Griffin. On the left (2.1a) is the original image. The isolated black segments of the Griffin are the “connected pieces” and the white enclosed areas are the “holes”. The other two (2.1b and 2.1c) have been colored in to highlight the 14 connected pieces and 4 holes respectively. Homology provides a precise mathematical description these structures.	12
2.2 Topological spaces X and Y . X consists of one connected line segment and Y has two disconnected line segments.	13
2.3 Topological spaces X_a, X_b, X_c , and X_d . Shading indicates that the enclosed area is filled.	13
2.4 The pattern κ described in Section 1.1. The homology of κ gives Betti numbers $\beta_0 = 1$ and $\beta_1 = 9$. True to the homology, we can easily count a single black connected component and nine holes, each of which is filled with a different color to illustrate this fact.	14

2.5	The left picture has a topological boundary given by $\{A\} \cup \{B\}$ (or $[A, B]$) and therefore does not form a loop (a 0-cube). The picture on the right, however, does not have a boundary (since we can't pick a definite start and end point) and therefore forms a loop (a 1-cube).	19
2.6	The boundary of the chain $[\widehat{A,B}] + [\widehat{B,C}] - [\widehat{C,D}] - [\widehat{D,A}]$ is zero in both pictures. The left picture, however, characterizes a loop (or hole) while the one on the right does not since it is "filled in" by 2-chain Q .	21
3.1	Patterns α , γ , and β at various thresholds. Some features of less stable patterns such as γ and β are lost when thresholded. Thresholds slightly higher than 128 tend to capture the features of the original pattern better.	27
3.2	A plot of the time series of Betti numbers for pattern γ . The zeroth Betti number β_0 is shown in red and the first Betti number β_1 shown in blue. Different thresholds $T = 112, 128, 144$, and 192 demonstrate the dramatic effect of thresholding on the calculation of Betti numbers for some patterns. For very high and low T , the image loses any resemblance to the original image (since it will appear mostly black or white). Slightly varying the threshold near 128, however, can help minimize the loss of information and remain truer to the original image. Interactive charts for each pattern type are available online at http://joelhawkins.info/thesis .	28
3.3	On the left is a spiral pattern with $(F, k) = (0.035, 0.060)$ while the figure on the right shows the pattern thresholded at $T = 144$. The Betti numbers for this pattern, as calculated by CHOMP, are $\beta_0 = 32, \beta_1 = 5$.	29
3.4	The probability P_i of the 20 most probable states s_i for pattern γ with $(F, k) = (0.022, 0.051)$. The entropy of this system is $S = 5.43427630$ nats. Interactive charts for each pattern type are available online at http://joelhawkins.info/thesis	31

- 4.1 A plot of entropy S for the systems described by discrete values of F, k for chemical V with $T = 144$. Adaptive resampling is implemented to increase the resolution of the plot beyond the initial 400 grid points. The phase diagram in Figure 1.3 is superimposed on the map to illustrate its agreement with Pearson's analysis. The system has higher entropy for F, k values in regions that Pearson identifies as least stable (near the bifurcation lines) [18]. 34
- 4.2 A plot of entropy S for the systems described by discrete values of F, k using only the initial 400 grid points. 4.2a and 4.2b show the entropy map for chemicals V and U respectively. Without resampling, the results still agree well with Pearson's analysis; the system has higher entropy near bifurcations where it is least stable. Furthermore, the entropy maps for each chemical species show near perfect agreement. 35
- 4.3 A plot of S over F, k space for both chemical V (4.3a) and U (4.3b) considering only time steps $[100, 2500]$ to remove the possible effects of initial transient states. The entropy for each chemical is slightly higher than that shown in Figure 4.2 where the initial 100 time steps are considered in the entropy calculation. 35
- 4.4 Entropy S of patterns γ, ϵ, β , and δ as the domain size n is increased. Intuitively, the entropy increases with the domain size since there is more space and therefore greater complexity. We notice that S grows faster for some patterns as n is scaled. Figure 4.5 shows pattern γ at the 2000th time step for each domain size n 37
- 4.5 The pattern γ at the 2000th time step as the domain size is varied. The spatial complexity increases greatly as n grows. A plot of the entropy S for each $n = 256, 512, 1024$ is given in Figure 4.4. 38
- 4.6 Three near-consecutive frames from a video of a SmoothLife simulation (before thresholding). The Betti numbers β_0, β_1 are given for reference. In this case, β_0 and β_1 consider white and black respectively. Computing the Betti numbers of every 3 frames gives $S = 6.82150758771$ which is highly entropic. Images adapted from S. Rafler, "Generalization of Conway's "Game of Life" to a continuous domain—SmoothLife," *arXiv preprint arXiv:1111.1567*, 2011.
Video available at <https://youtu.be/KJe9H6qS82I>. 38

Abstract

A method of topological analysis known as computational homology is explored in the context of the Gray-Scott reaction-diffusion model. Using the homology data of patterns generated by simulations of Gray-Scott (a time series of the Betti numbers), the Shannon entropy S is calculated over a large set of parameters to elucidate features about the system. The results of the calculation show strong qualitative agreement with previous nonlinear analysis of the Gray-Scott system. In addition, a formal mathematical description of homology is provided to uncover the roots of the theory. Other applications of homology, its viability as an analytic technique, and the problems encountered therein are also discussed.

For mom and dad.

Introduction

The study of pattern formation is incredibly diverse and certainly one of the most compelling aspects of nonlinear phenomenology. Scientists from many disciplines study pattern formation on scales ranging from that of the entire universe all the way down to the microscopic.¹ Just a cursory glance at the structure of a wind-swept sand dune, a snowflake, or even our own spiral galaxy reveals something interesting. Observation of these patterns might lead a scientist to ask what causes the pattern and wonder why there are patterns at all. This question gets complicated quickly because whether you see ‘God in the patterns’ or see them as the result of a non-equilibrium universe, there is still the question of what it means to have “structure” or “complexity” or even to be “interesting”.

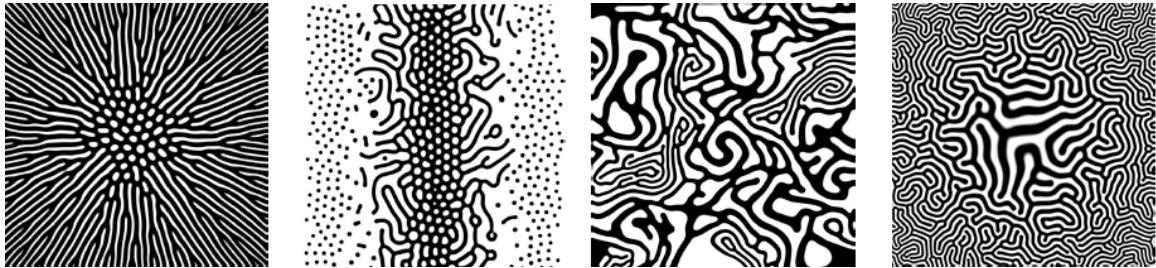


Figure 1: Patterns of the Gray-Scott reaction-diffusion system simulated by Karl Sims. Figure adapted from K. Sims, “Reaction-diffusion tutorial,” 2013. Available at <http://www.karlsims.com/rd.html>.

Of course, patterns in nature are inherently difficult to understand; they are often inhomogeneous, subject to many unknown forces, or simply too large or small to study carefully. So before arriving at conclusions about the structure of the universe, it is helpful to look at idealized systems. This can mean a tightly controlled experimental setup [3] or a completely computational model like the one discussed here; Figure 1 shows the patterns formed by simulations of the Gray-Scott chemical reaction-diffusion model [2]. Indeed, the vast amount of literature

¹See the introduction of Cross & Greenside for an overview of the study of pattern formation [1].

on the study of pattern formation looks to these simplified, yet no less dazzling, “prepared patterns” to draw conclusions about natural patterns [1]. One heavily studied example is Rayleigh-Bénard convection, the experimental setup for which is extremely accessible; a horizontal plane of fluid is heated from below and cooled from above which gives rise to patterns like the one shown in Figure 2.



Figure 2: An image of the “prepared pattern” formed by Rayleigh-Bénard convection. A horizontal plane of fluid is heated from below and cooled from above. Dark regions indicate the hot upflows and bright regions indicate cold downflows. Figure adapted from H. Kurtuldu, K. Mischaikow, and M. Schatz, “Measuring the departures from the Boussinesq approximation in Rayleigh-Bénard convection experiments,” *Journal of Fluid Mechanics*, vol. 682, pp. 543–557, 2011.

The study of patterns often comes down to the study of *image data*, especially in the setting of a computational simulation. There are many mathematical tools available to help interpret this kind of data but as the complexity of our information (i.e. the amount of data) increases, it becomes increasingly difficult to parse relevant information. Technological development in recent years not only makes capturing massive amounts of data possible, but commonplace. In the world of medical imaging, for example, data sets of X-Ray tomography, which allow for 3D reconstruction of biological structures such as the heart or lungs, can easily exceed dozens of gigabytes. While the increasing availability of data would serve only to improve our understanding of these systems, it is only as useful as our analytical methods allow. Traditional techniques may fall flat in the face of exceedingly sophisticated information.

Take for example the Fourier transform, a powerful method for analysis which is often used to remove noise or apply filters to images. We would expect the Fourier transform to provide some insight into the spatial frequency of the image²,

²In this case, the Fourier transform converts from the *spatial domain*, the image we see, to the *frequency domain*. The FT has been used for intricate pattern recognition, see [4].

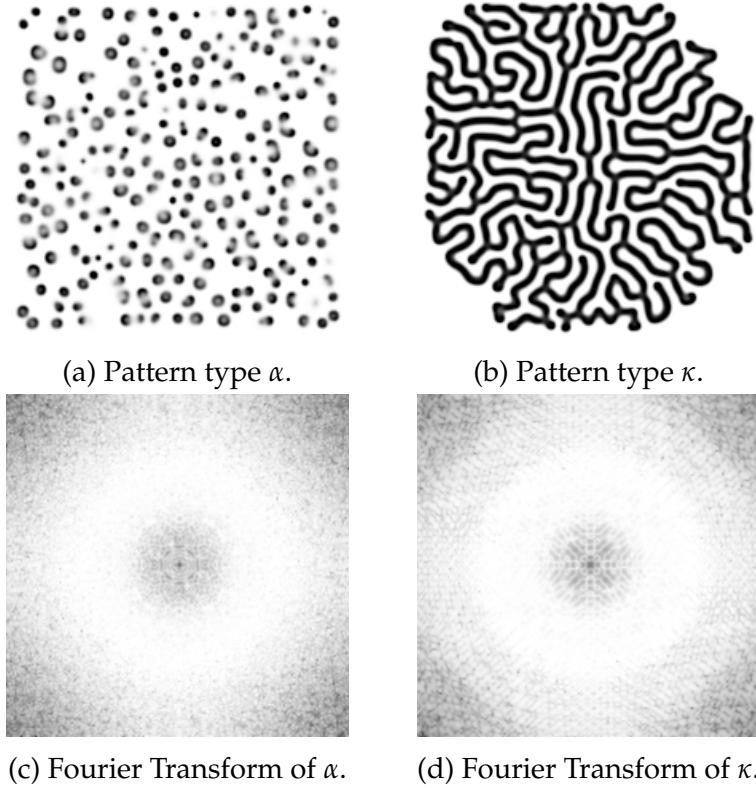


Figure 3: Two very distinct pattern types of the Gray-Scott reaction-diffusion system, α and κ (described further in Section 1.1). The Fourier transform of each pattern is hard to distinguish by eye and extracting meaningful information is difficult. The problematic nature of this method motivates our need for other, perhaps lower-level, analytic techniques.

but in some cases, this method fails to provide useful information. Examine the two distinct pattern types of the Gray-Scott reaction-diffusion system shown in Figure 3 (more about this system in Section 1.1). The two pattern types α and κ (Figures 3a and 3b respectively) are easy to differentiate by eye yet the Fourier transforms (Figures 3c and 3d) are disappointingly similar. Although this method is capable of extracting useful information, we would like some way to supplement our findings. The need for new methods arises and many times that means starting at the lowest level (i.e. the structures that make up the image) especially when the crucial information is geometric in nature.

But for many systems, it doesn't make sense to attempt to describe basic geometric structures in terms of the underlying mathematical equations (assuming we can write them down in the first place). This problem calls for a framework that gets at the geometric information even when faced with numerical error and minor perturbations. The theory of computational homology described in Chap-

ter 2 does just this. Under the umbrella of algebraic topology, homology provides a beautiful framework for transforming topology into algebra from which one can draw insight into global properties. Although homology has only recently been brought to the fore of experimental physics, its application has shown interesting results [5, 6, 7, 8]. This project explores the application of this theory to one pattern forming system in particular and highlights the information that may be derived from it.

Chapter 1

Reaction-Diffusion Systems

Reaction-diffusion (RD) systems are models that determine how concentrations of chemical species change in space and time. These systems are driven by two processes: chemical reaction and spatial diffusion. RD systems are governed by partial differential equations, the most basic of which might look something like

$$\frac{\partial u}{\partial t} = d\nabla^2 u + r(u). \quad (1.1)$$

This is sometimes called the Kolmogorov-Petrovsky-Piskounov equation in which u is a generic chemical species, d is a diffusion coefficient, $\nabla^2 u$ is the Laplace operator, and $r(u)$ is a general reaction term. Of course, RD systems consisting of a single chemical do not form interesting patterns since there is no reaction taking place.

RD systems are interesting because their solutions can show wide variety of complex patterns, many of which resemble patterns of nature such as spirals, stripes, and spots. One drawback of the simplicity of these systems is that quantitative comparison to experimental systems is difficult. Alan Turing, one of the first to study RD systems in detail, acknowledged this in the famous opening lines of his 1952 paper [9].

In this section a mathematical model of the growing embryo will be described. This model will be a simplification and an idealization, and consequently a falsification. It is to be hoped that the features retained for discussion are those of greatest importance in the present state of knowledge.

Despite this concession, reaction-diffusion systems constitute an important part of

the study of nonlinear dynamics today. Turing went on to suggest that reaction-diffusion systems of morphogens, chemicals that govern the pattern of embryo tissue development, may be able to explain the presence of spots or stripes on an organism. Although the science behind animal patterns is more complicated, Turing laid the framework by which patterns form from minor perturbations of otherwise homogenous systems. Since then, many others have noted the similarity between RD patterns and patterns in nature [10, 11, 12, 13, 14, 15, 16]. Figure 1.1 provides an example of how patterns formed by reaction-diffusion systems have been used to generate natural-looking textures in the context of computer graphics.

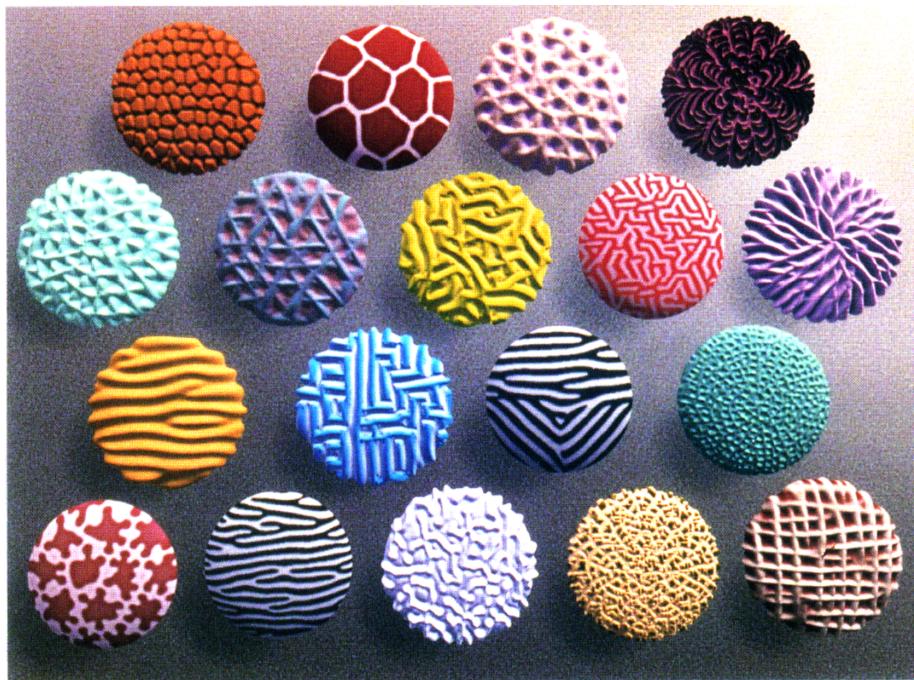


Figure 1.1: Patterns generated by reaction diffusion systems. Witkin compares these patterns to those of nature [16]; “Row 1: reptile, giraffe, coral, scalloped. Row 2: spiral, triweave, twisty maze, replication, purple thing. Row 3: sand, maze, zebra haunch, radial. Row 4: space giraffe, zebra, stucco, beats us.” Adapted from A. Witkin and M. Kass, “Reaction-diffusion textures,” *SIGGRAPH Computer Graphics*, vol. 25, no. 4, pp. 299–308, 1991.

1.1 The Gray-Scott model

One important model in the study of pattern formation is the Gray-Scott system which models the reaction of two generic chemical species, U and V [17]. The

model is based on the chemical reaction



where V is converted to an inert product, P , which doesn't interfere with the reaction of the system. V appears on both sides of the chemical reaction and thus catalyzes its own production. Gray and Scott developed the following non-dimensional partial differential equation (PDE) in which u and v represent the concentrations of chemicals U and V respectively.

$$\frac{\partial u}{\partial t} = d_u \nabla^2 u - uv^2 + F(1 - u) \tag{1.3}$$

$$\frac{\partial v}{\partial t} = d_v \nabla^2 v + uv^2 - (F + k)v \tag{1.4}$$

We see that both equations take the form of (1.1) only they are coupled. The boundary conditions are periodic and for simplicity, d_u , d_v , F , and k are taken to be constants. The first term in each equation, $d_u \nabla^2$ and $d_v \nabla^2$, are the diffusion terms. The Laplace operator, ∇^2 , is responsible for the diffusion of each chemical in space (like the diffusion of heat in the more familiar heat equation) while the *diffusion coefficients*, d_u and d_v , govern the diffusion rate. The $\pm uv^2$ terms are the *reaction terms* which convert U into V ; an increase in v is equal to the decrease in u , hence $+uv^2$ in (1.4). Since U will eventually get used up to generate V , the term $F(1 - u)$ is the *replenishment term* which reintroduces chemical U into the system (u has a maximum value of 1). Similarly, chemical V would increase without limit except for the *diminishment term*, $(F + k)v$, which serves to remove chemical V from the system. F is referred to the *feed rate* and determines the rate of replenishment while k is the difference between this rate and that of chemical V .

For some biological intuition, one can imagine the chemical reactions that occur in the development of an embryo as Turing theorized. In this case, the supply of chemicals might be the bloodstream where the replenishment and diminishment rates of the reaction are determined by the permeability of cell membranes.

The Gray-Scott system is particularly notable for the wide range of irregular patterns it produces. Previous analysis of the system by Pearson [18] identified at

least 12 different pattern types, all of which occur at different F, k with $d_u = 2d_v$.¹ Figure 1.2 shows the 12 quantifiably different patterns observed in this system determined by standard methods of nonlinear analysis (e.g. linear stability analysis and bifurcation theory) [19]. The chemical concentration of U is plotted over a 256×256 computational domain. The wide variety of observable patterns reveals the extremely variable behavior of this system as parameters are varied. Figure 1.3 provides a legend for the patterns, mapping each of them to their locations in k, F parameter space. One of the most compelling qualities of these patterns is their resemblance to patterns of nature, e.g. κ (Figure 1.2j) looks like coral and λ (Figure 1.2k) resembles the growth of bacteria. Other patterns, like β (Figure 1.2b), exhibit complex spatiotemporal behavior that resembles turbulence.

1.2 Numerical simulation

The system in (1.3–1.4) is solved by forward Euler integration of the discrete Laplacian obtained by the finite difference method,

$$\nabla^2 u(x, y) \approx u(x - 1, y) + u(x + 1, y) + u(x, y - 1) + u(x, y + 1) - 4u(x, y), \quad (1.5)$$

and similarly for v . In the Python programming language, this can be easily implemented using the Numpy package as below (see Appendix B.1 for full code).

```

Lu = ( U[0:-2,1:-1] + U[2:,1:-1] +
       U[1:-1,0:-2] + U[1:-1,2:] - 4*U[1:-1,1:-1] ) # Laplacian matrix for u
Lv = ( V[0:-2,1:-1] + V[2:,1:-1] +
       V[1:-1,0:-2] + V[1:-1,2:] - 4*V[1:-1,1:-1] ) # Laplacian matrix for v

uvv = u*v*v # corresponds to uv^2 term
u += (Du*Lu - uvv + F * (1-u) ) # concentration matrix for u
v += (Dv*Lv + uvv - (F+k)*v ) # concentration matrix for v

```

The matrices u and v contain concentration values for all points in the mesh. By iterating this calculation and plotting u and v as concentration maps, we can observe the evolution of the concentrations in time. The numerical accuracy this solution is limited by the discretization of the Laplacian in (1.5) which is second-order accurate.

¹Turing instabilities which give rise to spontaneous pattern formation cannot occur if all diffusion coefficients are equal. The ratio of 2 for diffusion coefficients has been found to show symmetry breaking for a wide range of parameter values [18].

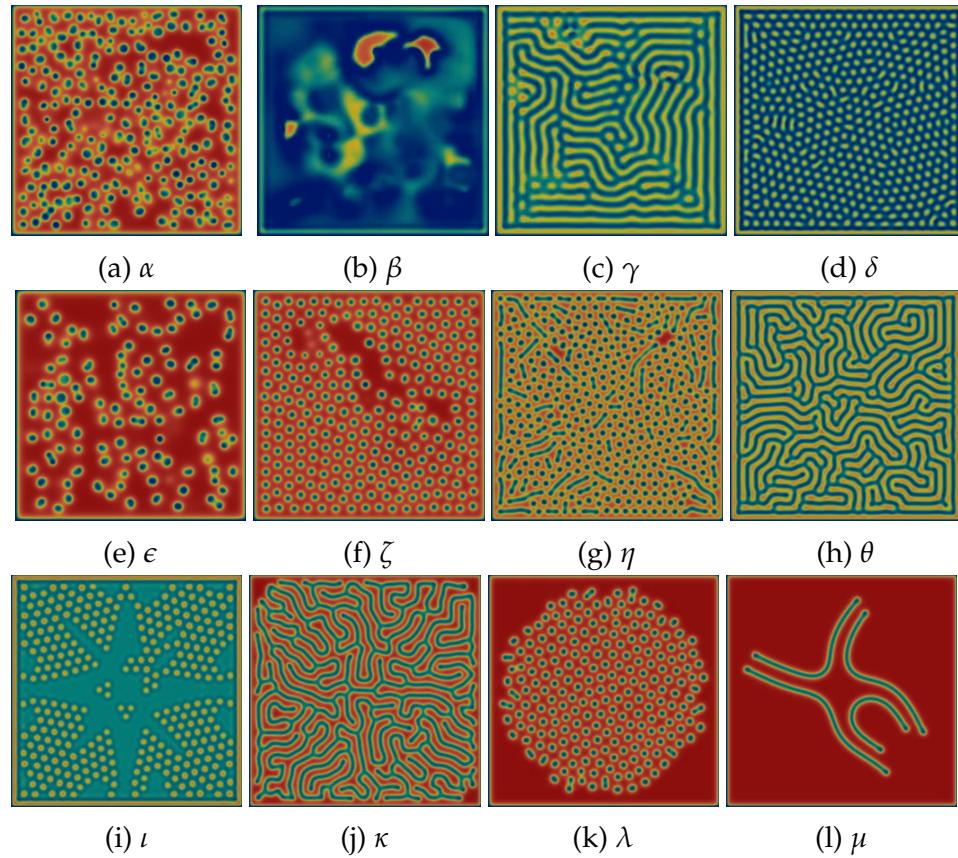


Figure 1.2: Patterns of chemical concentration U identified in [18]. Each pattern, Figure 1.2a—Figure 1.2l, is designated by a Greek letter which corresponds to the plot in Figure 1.3. Red and blue indicate $U = 1$ and $U \approx 0.2$ respectively. Note that a concentration plot of chemical V would appear as the inverse of U with red and blue swapped. Video simulations of each of these pattern types are available online at <http://joelhawkins.info/thesis>.

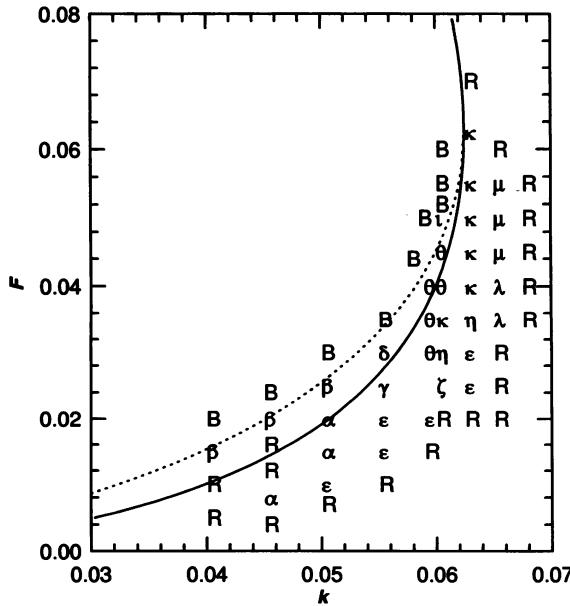


Figure 1.3: The mapping of Greek letters in Figure 1.2 to their location in k, F parameter space. R and B indicate that the system evolved to uniform red and blue states respectively. This figure also represents a phase diagram of the reaction kinetics. Between the solid and dotted line, the system is bistable for which there are two linearly stable steady states. As f passes below the dotted line, the non-trivial steady state becomes unstable through Hopf bifurcation giving stable periodic orbits for $k < 0.035$ and unstable ones for $k > 0.035$. The trivial state, ($U = 1, V = 0$) exists for all (f, k) outside the solid line. Adapted from J. E. Pearson, "Complex patterns in a simple system," *Science*, vol. 261, no. 5118, pp. 189–192, 1993.

A spatial grid of 256×256 points constitutes the mesh with a time step of 1^2 . The system was initialized with the state $U = 1, V = 0$ with a 40×40 area located symmetrically in the center perturbed with $U = 0.5, V = 0.25$. This square area is then further sprinkled with 1% random “noise” to catalyze the reaction. The patterns in Figure 1.2 were generated using this method and depict the concentration of chemical U . A plot of chemical V would appear as the inverse of U so only U is shown.

For each F, k , the simulation is run for 25,000 time steps. Every 10th image of the simulation is saved, so a total of 2,500 PNG files are produced to describe the pattern. The pattern images used for the analysis described in Chapter 3 use a greyscale colormap like that of Figure 3.

²There are no qualitative differences for domain sizes up to 1024×1024 and time steps as low as 0.01. Initial conditions also have little to no effect on the qualitative features of the resulting pattern after some time [18].

Chapter 2

Computational Homology

As large amounts of data become available, it becomes more difficult to determine what information is relevant. There are, of course, high and low-level approaches. A high-level approach like a fingerprint scanner or handwriting recognition might be the end-goal of one's analysis, but lower-level approaches like homology look at the geometric makeup of an object and are often a requisite step toward building higher-level processes. Homology is one way of analyzing *local* properties in order to extract information about *global* phenomena.

At this time, computational homology is a relatively new field and its application in physics has only recently been explored [20, 7, 3, 8]. Although homology is a field of algebraic topology, it combines the mathematics of several other fields including combinatorics and computation. The mathematical formalism behind homology is difficult to grasp so only the relevant information will be detailed in Section 2.1. For the interested reader, Section 2.2 on cubical homology presents a more thorough discussion of the mathematical background of the topic.

2.1 Homology overview

Homology is concerned with the enclosed *holes* and connected *pieces* in topological spaces. This vague statement might lead one to ask what exactly we mean by "connected pieces" and "holes". To gain an intuitive understanding, examine Figure 2.1 in which the connected pieces (the black segments) and holes (the white enclosed areas) of the Reed Griffin have been colored in. While it may seem easy enough to simply count these structures as I have done, the goal of homology is to provide a formal mathematical description of these geometric structures regardless of the

complexity or spatial dimension. Although the formalism of homology is difficult to understand, the relevant concepts are easily illustrated through examples. It's best to think about this in one dimension first.

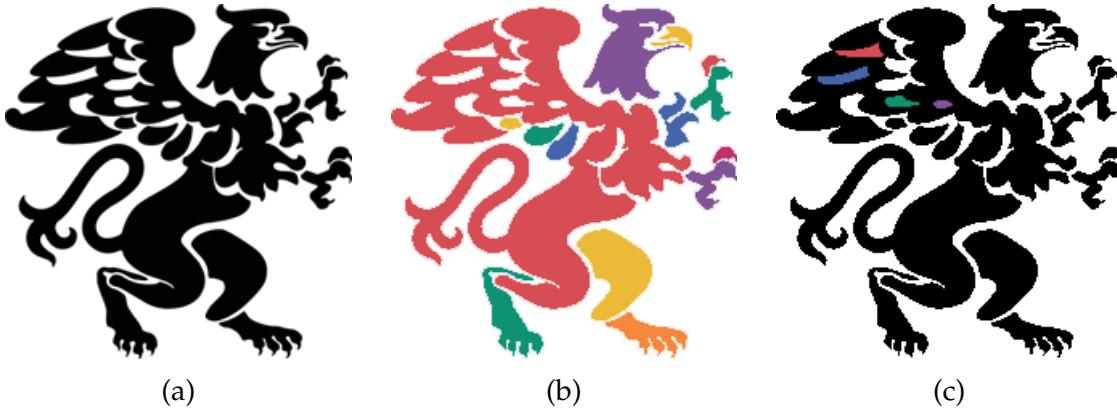


Figure 2.1: The homology of the Reed Griffin. On the left (2.1a) is the original image. The isolated black segments of the Griffin are the “connected pieces” and the white enclosed areas are the “holes”. The other two (2.1b and 2.1c) have been colored in to highlight the 14 connected pieces and 4 holes respectively. Homology provides a precise mathematical description these structures.

Figure 2.2 shows two simple topological spaces, X and Y . Although X and Y are spaces with one and two line segments respectively, in terms of homology one would say X consists of a single *connected piece* while Y has two distinct pieces. The fact that the line segments are straight or of different length is not important for the homology. In this one-dimensional example, the *zeroth homology groups* of each are

$$H_0(X) \cong \mathbf{Z}^1 \quad \text{and} \quad H_0(Y) \cong \mathbf{Z}^2, \quad (2.1)$$

where \mathbf{Z} is the group of integers. The homology pairs a topological space (e.g. X and Y) with an *abelian group*, a set of elements combined with operations that satisfy five axioms: closure, associativity, identity, invertibility, and commutativity (a full definition can be found in A.1.1). Notice, however, that the *zeroth homology group* of Y is \mathbf{Z}^2 ; the rank of the group, 2, is what accounts for the two distinct pieces, but more on this later.

Since there is a *zeroth homology group*, it makes sense that there would be a *first homology group*. Looking at the two-dimensional example in Figure 2.3, the homol-

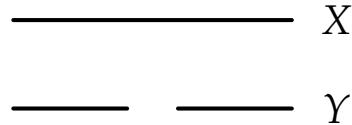


Figure 2.2: Topological spaces X and Y . X consists of one connected line segment and Y has two disconnected line segments.

ogy of each space X_a , X_b , X_c , and X_d is

$$H_0(X_a) \cong \mathbf{Z} \quad H_0(X_b) \cong \mathbf{Z} \quad H_0(X_c) \cong \mathbf{Z} \quad H_0(X_d) \cong \mathbf{Z}^2 \quad (2.2)$$

$$H_1(X_a) \cong \mathbf{Z} \quad H_1(X_b) \cong 0 \quad H_1(X_c) \cong \mathbf{Z} \quad H_1(X_d) \cong \mathbf{Z}. \quad (2.3)$$

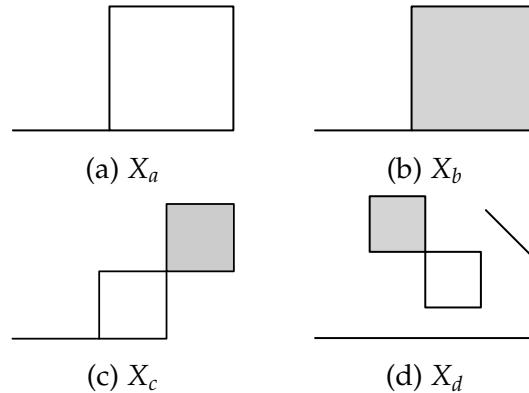


Figure 2.3: Topological spaces X_a , X_b , X_c , and X_d . Shading indicates that the enclosed area is filled.

Spaces X_a , X_b , and X_c have a *zeroth homology group* of \mathbf{Z} since there is a single connected component while X_d has \mathbf{Z}^2 to account for the two disconnected lines. In each space, a connected component forms an enclosed area (i.e. the squares). The square in Figure 2.3a forms a hole, a region completely enclosed by the black line. Figures 2.3a, 2.3c, and 2.3d each contain one hole. The shading in Figures 2.3b, 2.3c, and 2.3d indicates that the hole is filled and is thus no longer counted. Just as the *zeroth homology group* is concerned with connected segments, the *first homology group* is concerned with holes.

The terms “piece” and “hole” are informal. Formally, we could say that the k^{th} homology group, $H_k(X)$, represents the group of k -dimensional holes of X where a $k = 0$ hole is merely the gap between two components (e.g. Y in Figure 2.2). As I alluded to earlier, the the rank of the homology group (e.g. the rank 2 of \mathbf{Z}^2 in

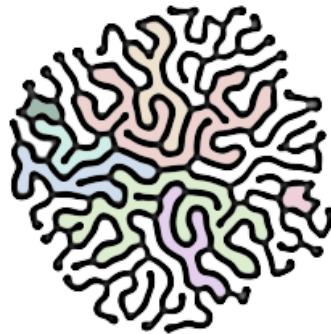


Figure 2.4: The pattern κ described in Section 1.1. The homology of κ gives Betti numbers $\beta_0 = 1$ and $\beta_1 = 9$. True to the homology, we can easily count a single black connected component and nine holes, each of which is filled with a different color to illustrate this fact.

(2.1)) represents the *number* of k dimensional holes.¹ This is called the *Betti number* β_k . Indeed, Betti numbers are non-zero for all $k < d$ where d is the dimension of the topological space. Betti numbers are the most important feature of the homology in this thesis since they assign a nice mathematical quantity to an otherwise visual characteristic of a topological space. Figure 2.4 gives the Betti numbers for pattern κ ; there is clearly one single connected component, shown in black, and nine enclosed holes which have been colored in. Thus, $\beta_0 = 1$ and $\beta_1 = 9$.

2.1.1 Prior work

Although the literature surrounding homology is relatively little, the techniques described here have been used to characterize complex patterns. Prior work by Gameiro and Mischaikow examines the 1D Gray-Scott system as well as the 2D FitzHugh-Nagumo model [7]. Using a time series of Betti numbers, they are able to calculate the maximal Lyapunov exponent (LE) which, if positive, implies existence of spatial-temporal chaos. They compare their computation of the LE using a time series of Betti numbers to that obtained through standard methods. The results are compelling for two reasons; one is that there is near-perfect agreement between the LE obtained through the Betti time series and standard methods which

¹The “dimension” of k is different from the dimension of the topological space. To describe the dimension of a space such as a cube in \mathbf{R}^3 , call it X , we write $\dim X = 3$. Structures of lower dimensions, such as the square faces that make up the cube, are *embedded* in the higher dimensional space X . For $k \geq \dim X$, $H_k(X) = 0$ since there are no structures embedded in a space with a higher dimension than that of the space itself. It is important to note that if we could place the topological spaces shown in Figure 2.3, which live in \mathbf{R}^2 on the page, into 3-dimensional space, \mathbf{R}^3 , this would not change the homology groups.

confirms that using homology data is an acceptable approach. The other is that, due to its topological features, the homology data is also able to capture spatial chaos which the LE calculated through the standard method does not (it only measures temporal chaos).

The work done by Gameiro, Mischaikow, and others provides a wonderful backdrop and inspiration for the analysis of the 2D Gray-Scott model in this thesis. The computational method for calculating homology data of this system is outlined in Chapter 3. In Section 3.2, I describe how this can be used to derive information about the dynamics of a system.

2.2 Cubical homology

In cubical homology, topological spaces are represented as a collection of cubes. This thesis is concerned with the interpretation of digital images as topological spaces. Digital images are quite literally a collection of two-dimensional cubes, *pixels*, thus a homology that examines these objects is a natural environment for examining the output of a computer simulation. In this section, I present a brief mathematical description of cubical homology that closely follows that of [21] and [5]. By skipping this section, one would miss some of the interesting subtleties of homology theory but a thorough understanding is by no means essential to this thesis. We'll start by defining elementary cubes, which make up the building blocks for the theory. It is important to keep in mind here that one of the fundamental ideas in homology theory is to connect topological objects (e.g. connected pieces and holes) to algebraic objects.

Definition 2.2.1. An *elementary interval* is an interval $I \subset \mathbb{R}$ of the form

$$I = [l, l+1] \quad \text{or} \quad I = [l, l]$$

for some $l \in \mathbb{R}$. To simplify notation, say

$$[l] = [l, l]$$

is an interval containing a single point, which we call a *degenerate* interval. Intervals of the form $[l, l+1]$ are called *nondegenerate*.

Definition 2.2.2. An *elementary cube* Q is a finite product of elementary intervals,

$$Q = I_1 \times I_2 \times \dots \times I_d \subset \mathbb{R}^d$$

where each I_i is an elementary interval. We denote the set of all elementary cubes in \mathbb{R}^d as \mathcal{K}^d .

The set of all elementary cubes, \mathcal{K} , is

$$\mathcal{K} := \bigcup_{d=1}^{\infty} \mathcal{K}^d.$$

Definition 2.2.3. Let $Q = I_1 \times I_2 \times \dots \times I_d \subset \mathbb{R}^d$ be an elementary cube. The *embedding number* of Q is defined to be d which we denote by $\text{emb}(Q)$. Interval I_i is the i^{th} component of Q and is written $I_i(Q)$. The *dimension* of Q is defined as the number of nondegenerate components in Q and denoted $\dim Q$. We refer to an elementary cube Q with $\dim Q = k$ as a *k-cube* and denote

$$\mathcal{K}_k := \{Q \in \mathcal{K} \mid \dim Q = k\},$$

and

$$\mathcal{K}_k^d := \mathcal{K}_k \cap \mathcal{K}^d.$$

The relationship between the embedding number and dimension might be a little muddy since it seems that they would always be the same. Observe that for elementary cube Q , if $\text{emb}(Q) = d$, then $Q \in \mathcal{K}^d$. The only general relation between the embedding number and dimension of Q is that

$$0 \leq \dim Q \leq \text{emb}(Q).$$

To illustrate this, imagine a Rubik's cube on a desk. The Rubik's cube itself has both $\text{emb}, \dim = 3$ while any one square *face* has $\text{emb} = 3$ but $\dim = 2$ (also see Example (2.2.1)).

Example 2.2.1. Given elementary cube $Q := [1, 2] \times [1, 2] \times [-1] \subset \mathbb{R}^3$, we have $I_1(Q) = [1, 2]$, $I_2(Q) = [1, 2]$, and $I_3(Q) = [-1]$ (which is degenerate). Therefore, $\text{emb}(Q) = 3$ and $\dim Q = 2$, due to the degenerate I_3 .

Now we must define the class of topological spaces for which we define the homology.

Definition 2.2.4. A set $X \subset \mathbb{R}^d$ is *cubical complex* if X can be written as a finite union of elementary cubes.

Given cubical complex $X \subset \mathbb{R}^d$, we define

$$\mathcal{K}(X) := \{Q \in \mathcal{K} \mid Q \subset X\}$$

and

$$\mathcal{K}_k(X) := \{Q \in \mathcal{K}(X) \mid \dim Q = k\}.$$

We can write $\mathcal{K}^d(X)$ to remind us that $X \subset \mathbb{R}^d$ as well as $\mathcal{K}_k^d := \mathcal{K}^d(X) \cap \mathcal{K}_k(X)$. For example, elements of $\mathcal{K}_0(X)$ are *vertices* of X , elements of $\mathcal{K}_1(X)$ are *edges* and so forth. $\mathcal{K}_k(X)$ are the *k-cubes* of X .

Once again, our goal is to establish a relationship between algebraic objects and topological spaces. The first step, then, is to associate some algebraic object with the elementary cubes that we just defined.

Definition 2.2.5. For each elementary *k-cube* $Q \in \mathcal{K}_k^d$, we associate an algebraic object \widehat{Q} which we call an *elementary k-chain* of \mathbb{R}^d where $\widehat{Q} : \mathcal{K}_k^d \rightarrow \mathbb{Z}$ is the function defined by

$$\widehat{Q}(P) = \begin{cases} 1 & \text{if } P = Q, \\ 0 & \text{otherwise.} \end{cases}$$

We also define $\widehat{0} : \mathcal{K}_k^d \rightarrow \mathbb{Z}$ to be the zero function, i.e. $\widehat{0}(Q) = 0$ for all $Q \in \mathcal{K}_k^d$. The set of all *elementary k-chains* of \mathbb{R}^d is given by

$$\widehat{\mathcal{K}}_k^d := \left\{ \widehat{Q} \mid Q \in \mathcal{K}_k^d \right\}$$

and the set of all *elementary chains* of \mathbb{R}^d is given by

$$\widehat{\mathcal{K}}^d := \bigcup_{k=0}^{\infty} \widehat{\mathcal{K}}_k^d.$$

As previously mentioned, the purpose of defining the *k-chain* \widehat{Q} versus *k-cube* Q is to bridge the gap between the algebra and the topology. Just as a *k-cube* describes a structure formed by the product of intervals (e.g. $[0, 1] \times [0, 1]$ is an elementary 2-cube), a *k-chain* describes a combination of *simplices* (a 0-simplex is a point, a 1-simplex is a segment, a 2-simplex is a triangle, and so on). In the context of a graph, a chain might describe a path between vertices (0-simplices) in the form

of a linear combination of the vertices.

For an elementary cube Q , we refer to \widehat{Q} as its *dual elementary chain*. Conversely, given elementary chain \widehat{Q} , we call Q its *dual elementary cube*. What we want is a one-to-one relationship between the elementary k -*cubes* (topological objects) and *elementary k-chains* (algebraic objects). In other words, the map of k -*cubes* (\mathcal{K}_k^d) to k -*chains* ($\widehat{\mathcal{K}}_k^d$) is a *bijection*.

Proposition 2.2.1. *The map $\phi : \mathcal{K}_k^d \rightarrow \widehat{\mathcal{K}}_k^d$ given by $\phi(Q) = \widehat{Q}$ is a bijection.*

Proof. See [5]. □

Proposition 2.2.1 allows us invoke the inverse of ϕ to go from an algebraic object, the elementary chain \widehat{Q} , to a topological set, Q . The following definition uses the algebra that we have built up to give the elementary k -*chains* algebraic structure.

Definition 2.2.6. The group C_k^d of k -*dimensional chains* (or k -*chains*) of \mathbb{R}^d is the free abelian group (see Appendix A.1.2) generated by the elementary chains of \mathcal{K}_k^d . Thus the elements of C_k^d are functions $c : \mathcal{K}_k^d \rightarrow \mathbb{Z}$ such that $c(Q) = 0$ for all but a finite number of elementary cubes $Q \in \mathcal{K}_k^d$. In particular, the set of elementary k -chains $\widehat{\mathcal{K}}_k^d$ is the basis for C_k^d . By the same notation used in Appendix A.1.1,

$$C_k^d := \mathbb{Z}(\mathcal{K}_k^d)$$

If $c \in C_k^d$, then $\dim c := k$.

Figure 2.5 illustrates how we can use information about the boundary to exact information about the k -cubes, but we are again using topological information (the existence of a boundary) to derive more topological information (the existence of loops). What we would really like to do is use algebra to get at this information, so we start by defining the algebraic boundary of a k -chain.

The set of elementary chains forms a basis for C_k^d , thus we would like to easily describe an arbitrary chain $c \in C_k^d$ in terms of the elements of $\widehat{\mathcal{K}}_k^d$. Definition (2.2.7) provides such a relation which is analogous to the dot product in vector space.

Definition 2.2.7. Consider $c_1, c_2 \in C_k^d$, where $c_1 = \sum_{i=1}^m \alpha_i \widehat{Q}_i$ and $c_2 = \sum_{i=1}^m \beta_i \widehat{Q}_i$ and α_i and β_i scalars. The *scalar product* of the chains c_1 and c_2 is defined as

$$\langle c_1, c_2 \rangle := \sum_{i=1}^m \alpha_i \beta_i.$$

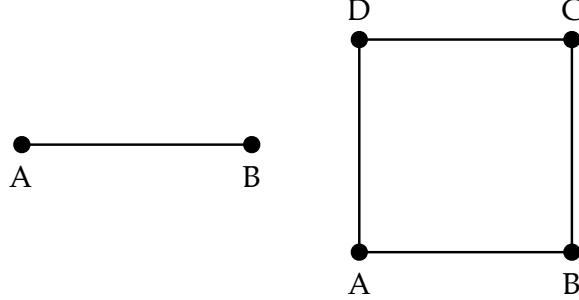


Figure 2.5: The left picture has a topological boundary given by $\{A\} \cup \{B\}$ (or $[A, B]$) and therefore does not form a loop (a 0-cube). The picture on the right, however, does not have a boundary (since we can't pick a definite start and end point) and therefore forms a loop (a 1-cube).

The astute reader will notice that this definition restricts us to describing a k -chain only in terms of k -dimensional cubes. We know, however, that cubes may be decomposed into lower dimensional faces. For example, the right picture in Figure 2.5 may be constructed from the four edges (or 1D faces) $[A, B]$, $[B, C]$, $[C, D]$, and $[D, A]$, and we would like to be able to write all k -chains in terms of lower-dimensional faces. This will be essential when the boundary operator is defined in 2.2.9 and provides the motivation for the following definition and proposition.

Definition 2.2.8. Given two elementary cubes $P \in \mathcal{K}_{k_1}^{d_1}$ and $Q \in \mathcal{K}_{k_2}^{d_2}$, let

$$\widehat{P} \diamond \widehat{Q} := \widehat{P \times Q}$$

This extends to arbitrary chains $c_1 \in C_{k_1}^{d_1}$ and $c_2 \in C_{k_2}^{d_2}$ by

$$c_1 \diamond c_2 := \sum_{P \in \mathcal{K}_{k_1}, Q \in \mathcal{K}_{k_2}} \langle c_1, \widehat{P} \rangle \langle c_2, \widehat{Q} \rangle \widehat{P \times Q}$$

The chain $c_1 \diamond c_2 \in C_{k_1+k_2}^{d_1+d_2}$ is called the *cubical product* of c_1 and c_2 .

Proposition 2.2.2. Let \widehat{Q} be an elementary cubical chain of \mathbb{R}^d with $d > 1$. Then there exist unique elementary cubical chains \widehat{I} and \widehat{P} with $\text{emb}(I) = 1$ and $\text{emb}(P) = d - 1$ such that

$$\widehat{Q} = \widehat{I} \diamond \widehat{P}$$

Proof. See [5]. □

Definition 2.2.9. Given $k \in \mathbb{Z}$, the *cubical boundary operator* or *cubical boundary map* given by

$$\partial_k : C_k^d \rightarrow C_{k-1}^d$$

is a homomorphism of free abelian groups, which is defined for an elementary chain $\widehat{Q} \in \widehat{\mathcal{K}}_k^d$ by induction on the embedding number d as follows. Consider first the case $d = 1$. Then Q is an elementary interval and hence $Q = [l] \in \mathcal{K}_0^1$ or $Q = [l, l+1] \in \mathcal{K}_1^1$ for some $l \in \mathbb{Z}$. Define

$$\partial_k \widehat{Q} := \begin{cases} 0 & \text{if } Q = [l], \\ [\widehat{l+1}] - [\widehat{l}] & \text{if } Q = [l, l+1] \end{cases}$$

Now assume that $d > 1$. Let $I = I_1(Q)$ and $P = I_2(Q) \times \dots \times I_d(Q)$. Then by 2.2.2,

$$\widehat{Q} = \widehat{I} \diamond \widehat{P}.$$

Define,

$$\partial_k \widehat{Q} := \partial_{k_1} \widehat{I} \diamond \widehat{P} + (-1)^{k_1} \widehat{I} \diamond \partial_{k_2} \widehat{P},$$

where $k_1 = \dim I$ and $k_2 = \dim P$. Finally, we extend the definition to all chains by linearity; that is, if $c = \alpha_1 \widehat{Q}_1 + \alpha_2 \widehat{Q}_2 + \dots + \alpha_m \widehat{Q}_m$, then

$$\partial_k c = \alpha_1 \partial_k \widehat{Q}_1 + \alpha_2 \partial_k \widehat{Q}_2 + \dots + \alpha_m \partial_k \widehat{Q}_m.$$

The domain of ∂_k is the k -chains, so if we know that $c \in C_k^d$, it is redundant and labor intensive to write the subscript k so we simplify to ∂ . Geometrically speaking, the boundary of a k -chain is simply the alternating sum of its $(k-1)$ -dimensional faces. As Figure 2.6 demonstrates, however, merely having a boundary sum equal to zero is not enough to constitute a loop. The right picture does not characterize a hole since it is filled in by the 2-chain Q .² This boundary is represented algebraically by $\partial(\widehat{Q}) = [\widehat{A, B}] + [\widehat{B, C}] - [\widehat{C, D}] - [\widehat{D, A}]$. We would represent the boundary topologically by $[A, B] \cup [B, C] \cup [C, D] \cup [D, A]$.

We can now say that holes are characterized by chains that have a boundary equal to zero, but are not themselves the boundary of other chains. In order to count the holes, then, we must count the chains which have zero boundary, but are not boundaries. The following definitions help us achieve this.

²Not to be confused with 2 Chainz [22].

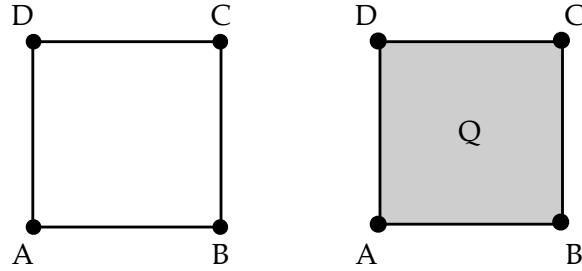


Figure 2.6: The boundary of the chain $[\widehat{A}, \widehat{B}] + [\widehat{B}, \widehat{C}] - [\widehat{C}, \widehat{D}] - [\widehat{D}, \widehat{A}]$ is zero in both pictures. The left picture, however, characterizes a loop (or hole) while the one on the right does not since it is “filled in” by 2-chain Q .

Definition 2.2.10. Let $X \subset \mathbb{R}^d$ be a cubical complex. Let $\widehat{\mathcal{K}}_k(X) := \{\widehat{Q} \mid Q \in \mathcal{K}_k(X)\}$. We define the set of k -chains of X as the subgroup $C_k(X)$ of C_k^d generated by the elements of $\widehat{\mathcal{K}}_k(X)$.

Proposition 2.2.3. Let $X \subset \mathbb{R}^d$ be a cubical complex. Then

$$\partial_k(C_k(X)) \subset C_{k-1}(X)$$

Proof. See [5]. □

This leads to the following definition.

Definition 2.2.11. The boundary operator of the cubical complex X is defined to be

$$\partial_k^X : C_k(X) \rightarrow C_{k-1}(X)$$

obtained by restricting $\partial_k : C_k^d \rightarrow C_{k-1}^d$ to $C_k(X)$.

An extremely important property of the boundary operator is defined in the following proposition.

Proposition 2.2.4.

$$\partial \circ \partial = 0$$

Proof. See [5]. □

It should make sense that if we are to take the boundary of a topological object, the boundary itself should have a lower embedding number (the boundary of a square, $k_1 = 2$, is made up of lines, $k_2 = 1$). It may not seem immediately intuitive that the boundary of a boundary is zero; the boundary of a disk is a circle

which has boundary equal to zero; the boundary of a baseball is its spherical shell which also has a zero boundary. At this point we are tantalizingly close to defining homology—but wait! The following descriptions will be helpful in a moment.

Definition 2.2.12. Let $X \subset \mathbb{R}^d$ be a cubical set. A k -chain $z \in C_k(X)$ is called a k -cycle in X if $\partial z = 0$. Thus the set of all k -cycles of X is the kernel of ∂_k^X and so it is a subgroup of $C_k(X)$. We denote the set of all k -cycles by $Z_k(X)$. In short,

$$Z_k(X) := \ker \partial_k^X = C_k(X) \cap \ker \partial_k \subset C_k(X). \quad (2.4)$$

A k -chain $z \in C_k(X)$ is called a *boundary* in X if there exists a $(k+1)$ -chain $c \in C_{k+1}(X)$ such that $\partial c = z$. Thus the set of all boundary elements in $C_k(X)$ is the image³ of ∂_{k+1}^X , and so it is also a subgroup of $C_k(X)$. We denote the set of all boundary elements in $C_k(X)$ by $B_k(X)$. Once again,

$$B_k(X) := \text{im } \partial_{k+1}^X = \partial_{k+1}(C_{k+1}(X)) \subset C_k(X). \quad (2.5)$$

We can now say more precisely what we wanted before; we want to characterize holes by chains that have zero boundary (i.e. k -cycles, the elements of $Z_k(X)$) but are not themselves the boundary of other chains. The set of all k -chains that represent boundaries of other chains is $B_k(X)$, so we want to count the elements of $Z_k(X)$ that are not in $B_k(X)$, and this will define our set of k -dimensional holes. This is easily done by taking the quotient group of $Z_k(X)$ by $B_k(X)$ but this requires that $B_k(X)$ be a subgroup of $Z_k(X)$. By Proposition 2.2.4, $\partial c = z$ implies $\partial z = \partial^2 c = 0$, therefore every boundary is a cycle and $B_k(X)$ is a subgroup of $Z_k(X)$ so we may rightfully proceed to the most important definition of this section.

Definition 2.2.13. The k^{th} *cubical homology group* of X is the quotient group

$$H_k(X) := Z_k(X) / B_k(X).$$

The homology of X is the collection of all homology groups of X . The shorthand for this is

$$H_*(X) := \{H_k(X)\}_{k \in \mathbb{Z}}.$$

³That's "image" in the mathematical sense (i.e. the subset that contains the output of a function). The shorthand for this is $\text{im } f$ for function f .

For a cubical set $X \subset \mathbb{R}^d$ we can show that, for $i = 0, \dots, d - 1$,

$$H_i(X) = \mathbb{Z}^{\beta_i} \oplus \mathbb{Z}_{b_1} \oplus \mathbb{Z}_{b_2} \oplus \cdots \oplus \mathbb{Z}_{b_k},$$

where β_i is a nonnegative integer, \mathbb{Z}_b is the group of integers modulo b , $b_i > 1$ provided $k > 0$, and b_i divides b_{i+1} for $i \in \{1, 2, \dots, k-1\}$ provided $k > 1$. For $i \geq d$ we have $H_i(X) = 0$.

Integer β_i is known as the i^{th} *Betti number* of X and b_1, b_2, \dots, b_k are the *torsion coefficients* of $H_i(X)$. In general, $\beta_i := \text{rank}(H_i(X))$. Spaces with dimension $d \leq 3$ do not have torsion coefficients, just $H_i(X) = \mathbb{Z}^{\beta_i}$, so we need not worry about them for our purposes [21].

This is certainly a great abstraction from what we started with before, connected pieces and holes in a topological space, but here is some geometrical intuition. As previously indicated, the Betti numbers encode some geometrical information. β_0 is equal to the number of connected pieces of X , β_1 is the number of holes (or loops) if $d = 2$ or the number of tunnels if $d = 3$. β_2 equals the number of cavities if $d = 3$.

Example 2.2.2. An ordinary bike tube (a torus) has $\beta_0 = 1$, the single connected piece of rubber; $\beta_1 = 1$, one hole in the center; and $\beta_2 = 1$, the hollow cavity inside the tube.

The mathematical tour-de-force that we've just undertaken might seem like major overkill. After all, here we are merely concerned with counting geometric structures that could theoretically be eyeballed (tedious as that may be). I argue, however, that the homology maintains some nice features for us. For one, it provides a mathematically rigorous definition of the structures in question. Furthermore, the homology of any structure is unchanged in any higher-dimensional space (e.g. the homology groups of an empty square are the same in \mathbf{R}^2 as in \mathbf{R}^3 or \mathbf{R}^4 for that matter). Homology is not concerned with size or shape of any object either; the homology of a coffee mug is identical to that of Figure 2.3a. The theory reduces the amount of information required to describe an object to a few topological quantities which may be difficult to grasp visually. Analysis of higher dimensional data, such as a 4D construction of medical imaging data, which would require a great amount of thought to assess visually, is completely feasible through cubical homology [5]. Due to its dimension-independent formulation, the applications of cubical homology are limited only by the ability to construct sensible

topological information. In the following chapter, we shall see how the homology theory described here can be applied to analyze patterns of the Gray-Scott system.

Chapter 3

Methods and Procedures

In this section, the methods for extracting homology information, i.e. Betti numbers, from images generated by simulating the Gray-Scott system are outlined. Section 3.1.1 describes the process of preparing the images for computation and the considerations and problems that arise. In Section 3.2, the Betti numbers are used to calculate the entropy of the Gray-Scott system as parameters are varied. For information on how the pattern images were generated in the first place, see Section 1.2.

3.1 Obtaining Betti numbers

Given the complicated overview of cubical homology theory given in Section 2.2, one might expect extracting the Betti numbers from an image to be a difficult undertaking. Fortunately, CHOMP, (Computational Homology Project), a homology software package developed by the group of Konstantin Mischaikow (Mathematics '79) at Rutgers University (formerly at Georgia Tech), facilitates this process [23]. Furthermore, the way it works is extremely intuitive, essentially counting clusters of adjacent pixels. CHOMP requires a 2-bit binary image as input.¹ Since the images output by the Gray-Scott simulation are greyscale (see Section 1.2), they must first be converted to a 2-bit image.

¹That is, an image with *only* black and white pixels.

3.1.1 Thresholding

The output of the Gray-Scott simulation is a series of 8-bit greyscale images, i.e. there are 256 possible shades of grey (0 is black, 255 is white). The color map is such that concentrations $V_{min} = 0.0 \rightarrow 255$ and $V_{max} = 0.4 \rightarrow 0$. In other words, white and black indicate low and high concentration of V respectively.² Each image is then thresholded at some value $T \in [0, 255]$. That is, all pixels with intensity $< T$ are now black and those with intensity $> T$ are now white.

A logical choice for T is the median pixel intensity of the image [20]. But, in the case of sparse patterns like α and ϵ (Figures 1.2a and 1.2e respectively), this results in a completely black image since the median is very high. Although other adaptive methods of thresholding exist [6], the definitive answer to thresholding problems is *persistent homology*, a more sophisticated computational homology technique which is a large leap in terms of complexity and outside the scope of this thesis [24]. This approach is concerned with the “birth” and “death” of homology components as the threshold is varied in a way that circumvents the need for a threshold altogether. Persistent homology has recently seen great success in analyzing large sets of nonlinear data [25].

Short of persistent homology, another reasonable choice would be to split right down the middle, $T = 128$, but as Figures 3.1 and 3.2 illustrate, some information can be lost in the process. Thus the optimal choice of threshold depends entirely on the characteristics of the image. The patterns produced by the Gray-Scott system are varied and no single value of T is ideal for all (F, k) , but empirically, T that is near 128 better agrees with the characteristics of the original pattern. After experimentation, the value $T = 144$ was chosen to perform all the calculations for Gray-Scott patterns of chemical V .

Provided in the CHOMP software package is a method for simply thresholding images, `chomp-greyscale-to-cubical` (which takes a single image input, a threshold value, and an output filename). The output is a text file which contains the coordinates of white pixels.

² V_{max} was chosen based on the average maximum of $V \approx 0.4$ for all the patterns examined.

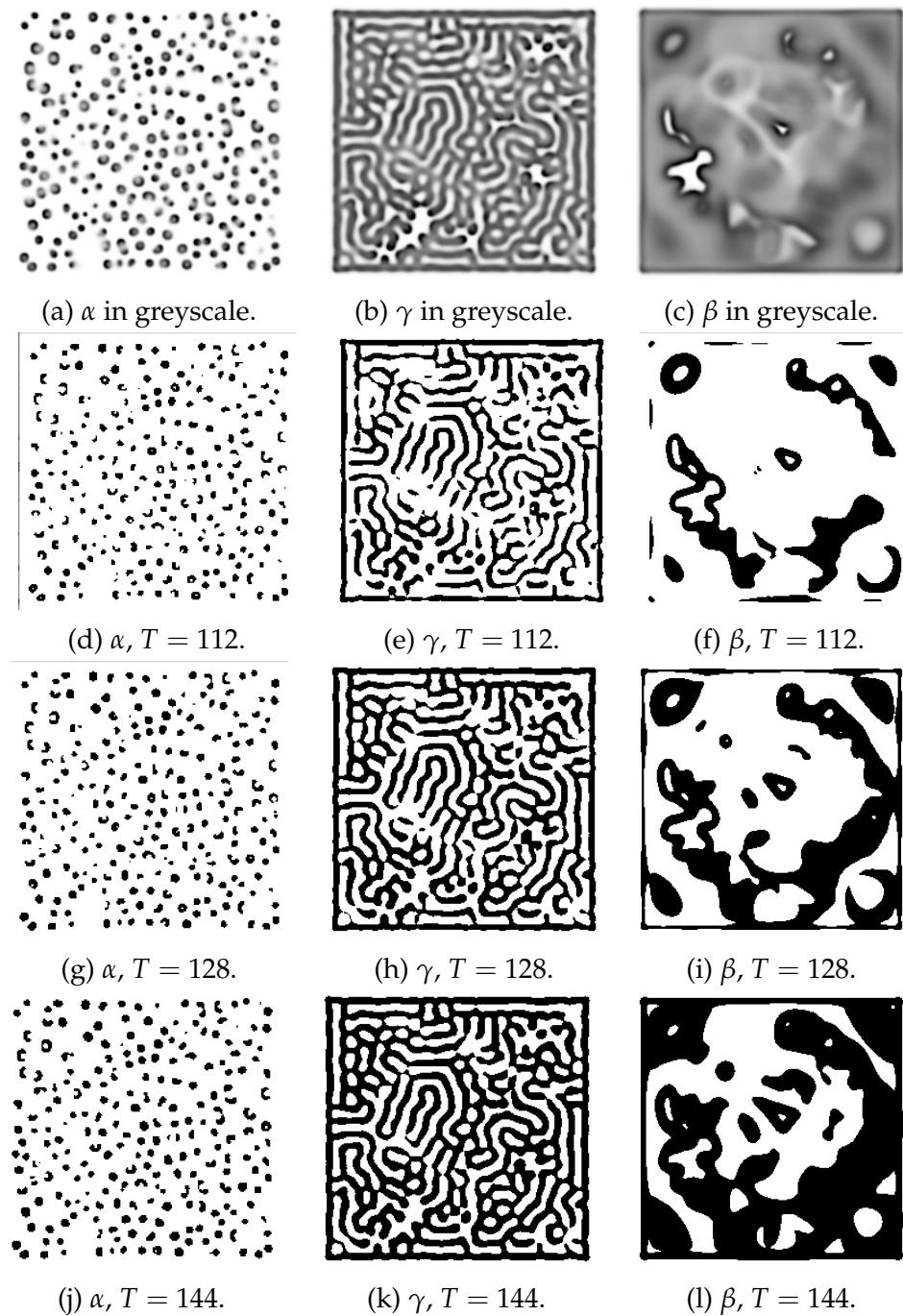


Figure 3.1: Patterns α , γ , and β at various thresholds. Some features of less stable patterns such as γ and β are lost when thresholded. Thresholds slightly higher than 128 tend to capture the features of the original pattern better.

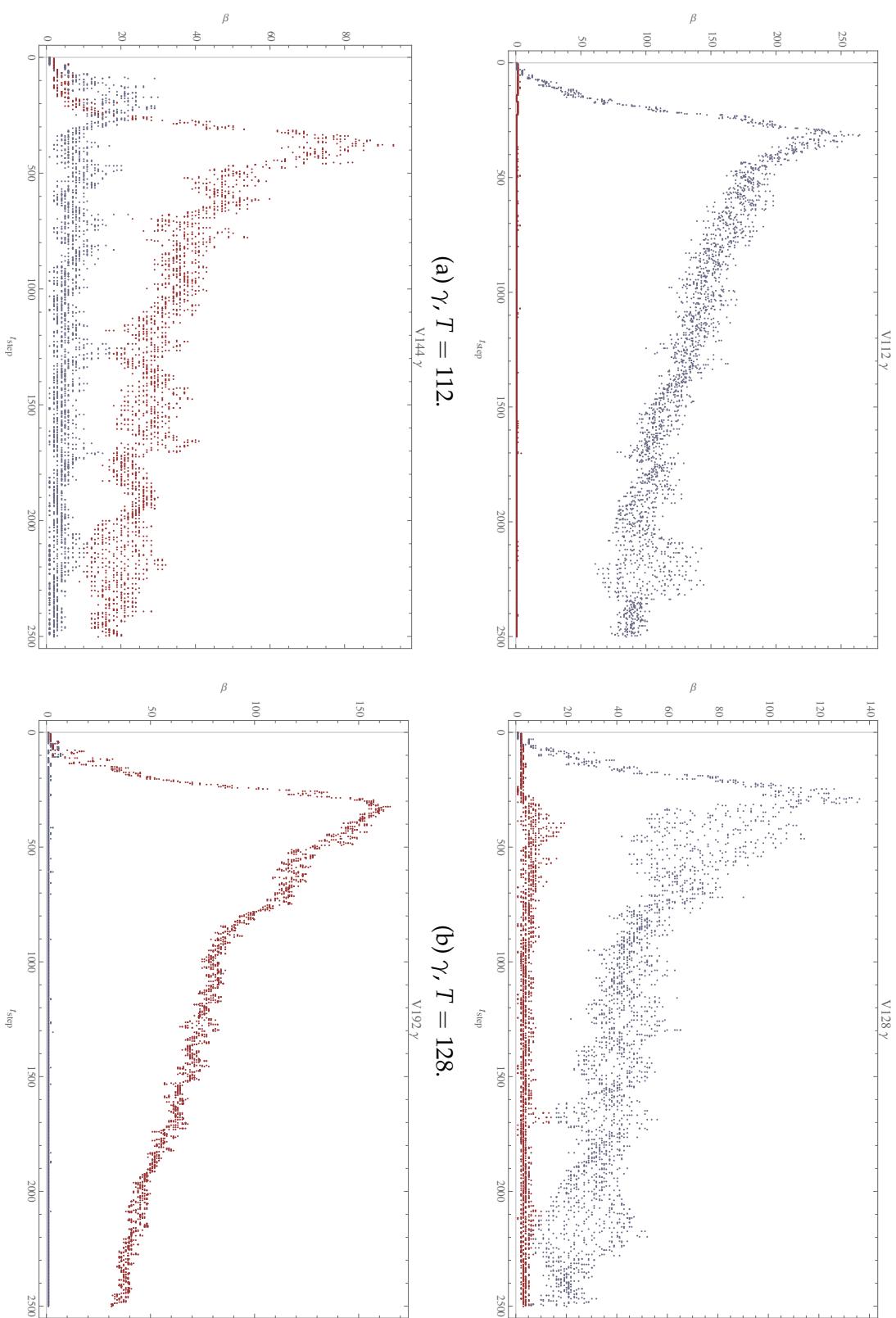


Figure 3.2: A plot of the time series of Betti numbers for pattern γ . The zeroth Betti number β_0 is shown in red and the first Betti number β_1 shown in blue. Different thresholds $T = 112, 128, 144$, and 192 demonstrate the dramatic effect of thresholding on the calculation of Betti numbers for some patterns. For very high and low T , the image loses any resemblance to the original image (since it will appear mostly black or white). Slightly varying the threshold near 128 , however, can help minimize the loss of information and remain truer to the original image. Interactive charts for each pattern type are available online at <http://joelhawkins.info/thesis>.

3.1.2 Computational homology

Once the images are thresholded at some value, the CHOMP method chomp-cubical processes and returns Betti numbers β_0 , β_1 , and β_2 . A single calculation of Betti numbers takes about 1-3s on a 4.2 GHz Intel i7 processor constituting the greatest bottleneck in the process. Figure 3.3 shows the intermediate stages of calculating Betti numbers for a single choice of F, k values. This process is performed for each of 2,500 text files produced by thresholding to generate a single CSV file of the Betti numbers β_0 , β_1 , and β_2 for each time step.³

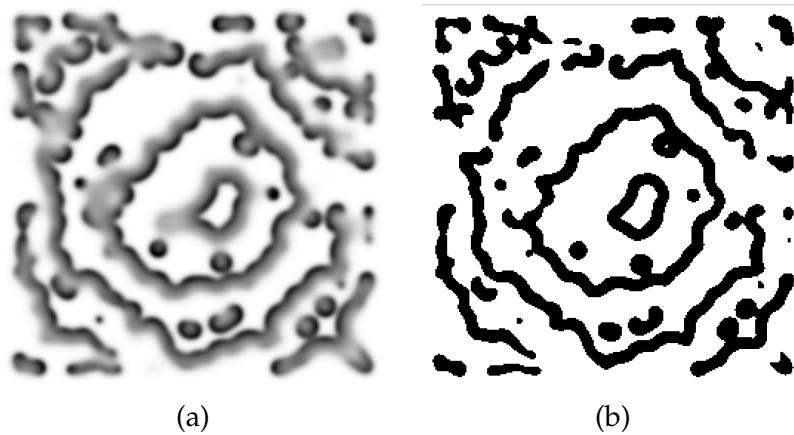


Figure 3.3: On the left is a spiral pattern with $(F, k) = (0.035, 0.060)$ while the figure on the right shows the pattern thresholded at $T = 144$. The Betti numbers for this pattern, as calculated by CHOMP, are $\beta_0 = 32$, $\beta_1 = 5$.

3.2 Calculating entropy

In physics, entropy usually denotes the amount of “disorder” of a system. Shannon’s entropy, $S(X)$, indicates the average amount of information that an observer gains *after* measuring a realized outcome x of the random variable X [26]. We wish to use homology information to provide a sense of how predictable (how complex) the dynamics of the Gray-Scott system are for a given choice of parameters (F and k). In general, the Shannon entropy S of some variable X with possible values

³The Betti number β_2 is included only for posterity; $\beta_2 = 0$ for all time steps since the images are only 2D.

$\{x_1, \dots, x_N\}$ and probability distribution $P(x_i) = P_i$ is defined by

$$S(X) = - \sum_{i=1}^N P_i \log P_i. \quad (3.1)$$

In our case, the Shannon entropy gives a picture of the average minimum number of topological “states” required to describe the system based on the frequency of the states explored by the system (i.e. how often some state occurs in the time series).⁴ A *state* in this case is taken to be a unique pair of Betti numbers $s_i = \{\beta_0, \beta_1\}_i$ at the i th time step. Although s_i in general doesn’t describe a *unique* pattern (any two states such that $s_i = s_j$ could look very different), it captures the fundamental topology of the system at that moment. Furthermore, the set of states within a given set of parameters (any F, k) is more meaningful. For example, if we observe the topology of state s_i for pattern α , then we can make an informed prediction as to what some other state s_j might look like for that pattern (assuming we know what the characteristic/steady-state pattern for α looks like). In general, we would expect higher entropy for more dynamic, complex patterns.

For N total (non-unique) states equal to the number of time steps, the probability P_i of state s_i given N_i , the number of times state s_i occurs, is simply

$$P_i = \frac{N_i}{N}. \quad (3.2)$$

Figure 3.4 charts the twenty states, s_i , with the highest probability, P_i (i.e. the twenty most probable states) for the pattern γ (Figures 1.2c and 3.1b). More dynamic patterns have a more even distribution of many states, each with lower P_i . Patterns like μ (Figure 1.2l) that evolve slowly have a very small number of possible states each with high P_i and therefore a very low entropy.

Since we can use homology information, namely the time series of Betti numbers, to calculate the entropy of a single pattern, we might ask if we can use this information to gain insight into the dynamics of the entire Gray-Scott pattern-forming system. In the following chapter, we examine how the entropy changes as parameters are varied, how it compares to Pearson’s analysis, and investigate other information that can be derived from the time series of Betti numbers.

⁴The units of S are called “nats” when the log in (3.1) is the natural logarithm.

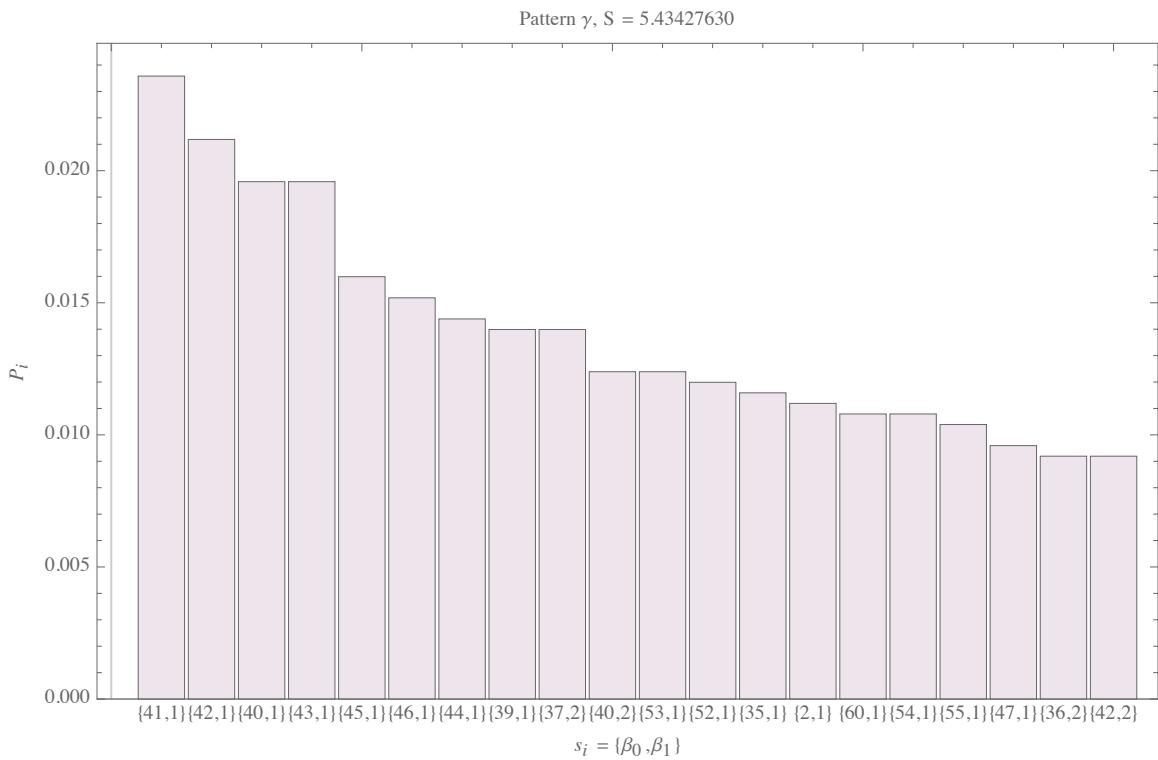


Figure 3.4: The probability P_i of the 20 most probable states s_i for pattern γ with $(F, k) = (0.022, 0.051)$. The entropy of this system is $S = 5.43427630$ nats. Interactive charts for each pattern type are available online at <http://joelhawkins.info/thesis>.

Chapter 4

Results

4.1 Entropy maps

Looking back at the analysis of the Gray-Scott system by Pearson, in particular Figure 1.3, the least stable regions of F, k occur near the bifurcation lines [18]. We would expect that the most complex patterns with F, k in this region would have higher entropy. In order to elucidate information about the dynamics of the Gray-Scott system, (3.1) is calculated for all $\{F, k \mid F \in [0.004, 0.08], k \in [0.03, 0.07]\}$ in an evenly spaced 20×20 grid (with $dF = 0.004$ and $dk \approx 0.002$) corresponding to the domain of Pearson's map of F, k parameter space (Figure 1.3). The diffusion coefficients are fixed at $d_u = 0.16, d_v = 0.08$ for all points. There are then 400 initial values of F, k for which the entropy S is calculated.

In order to increase the resolution, an adaptive resampling method is implemented in the following manner. For any F, k pair of the original 400 points with $S > 0.5$, the entropy is calculated for the four adjacent points $(F \pm dF/2, k)$ and $(F, k \pm dk/2)$ to form a 5 point stencil around the point. This of course requires starting from the Gray-Scott simulation outlined in Section 1.2 and performing the homology calculations for each new F, k pair.

The results of the entropy calculation agree well with our expectations. The plot in Figure 4.1 shows the entropy S of the Gray-Scott system for chemical V as F, k is varied. Note that systems of higher S , the lighter regions of the plot, occur more densely near the dotted line which indicates the Hopf bifurcation. Figure 4.2 shows the entropy S over the same domain for chemicals V and U , respectively, without adaptive sampling (only the initial 400 grid points). These too agree with Pearson's analysis and show that both chemicals U and V exhibit similar dynamics.

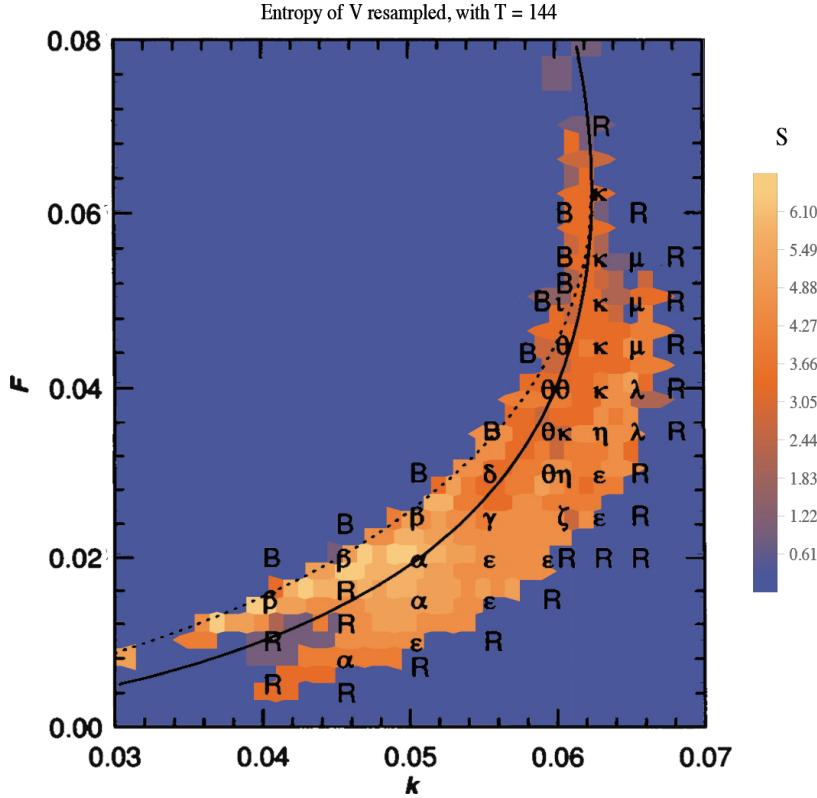


Figure 4.1: A plot of entropy S for the systems described by discrete values of F, k for chemical V with $T = 144$. Adaptive resampling is implemented to increase the resolution of the plot beyond the initial 400 grid points. The phase diagram in Figure 1.3 is superimposed on the map to illustrate its agreement with Pearson's analysis. The system has higher entropy for F, k values in regions that Pearson identifies as least stable (near the bifurcation lines) [18].

With 2,500 time steps for each F, k , producing a map of the entropy over F, k space requires over 1 million calls to chomp-cubical.¹ Computed with eight parallel processes on a 4.2GHz Intel i7 processor, this takes about 3-4 days of computation time.

4.2 Transient states

One consideration in these calculations is the existence of initial transient states. As the Gray-Scott simulation runs, there are about 100 time steps in which the system has not yet reached its steady-state. The entropy maps in Figure 4.3 consider only the time steps [100, 2500], ignoring the transient states which might affect

¹Computing the Betti numbers of the system is by far the slowest operation and significantly bottlenecks the processing time.

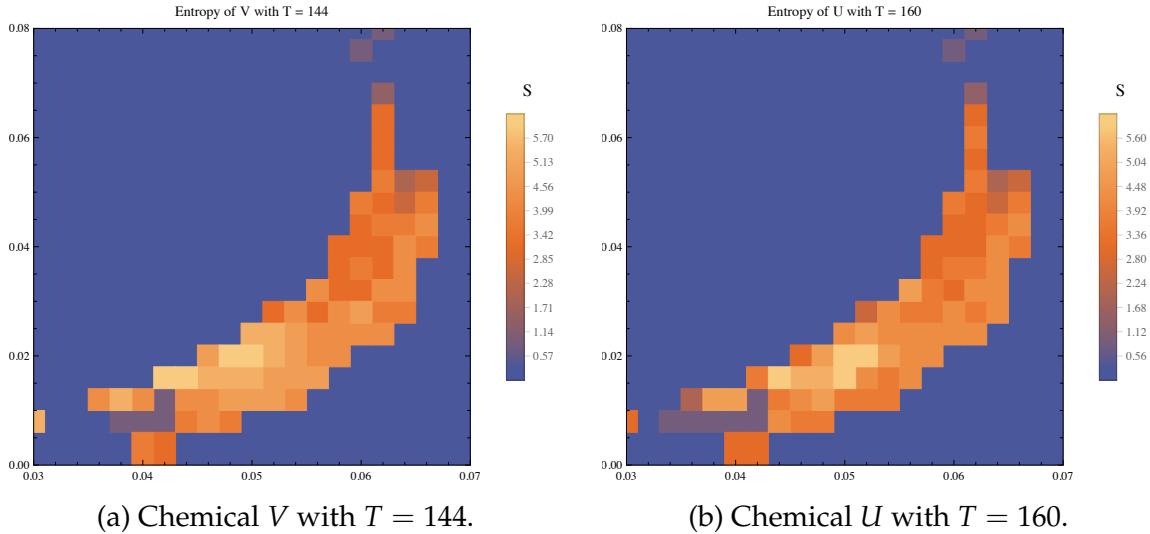


Figure 4.2: A plot of entropy S for the systems described by discrete values of F, k using only the initial 400 grid points. 4.2a and 4.2b show the entropy map for chemicals V and U respectively. Without resampling, the results still agree well with Pearson's analysis; the system has higher entropy near bifurcations where it is least stable. Furthermore, the entropy maps for each chemical species show near perfect agreement.

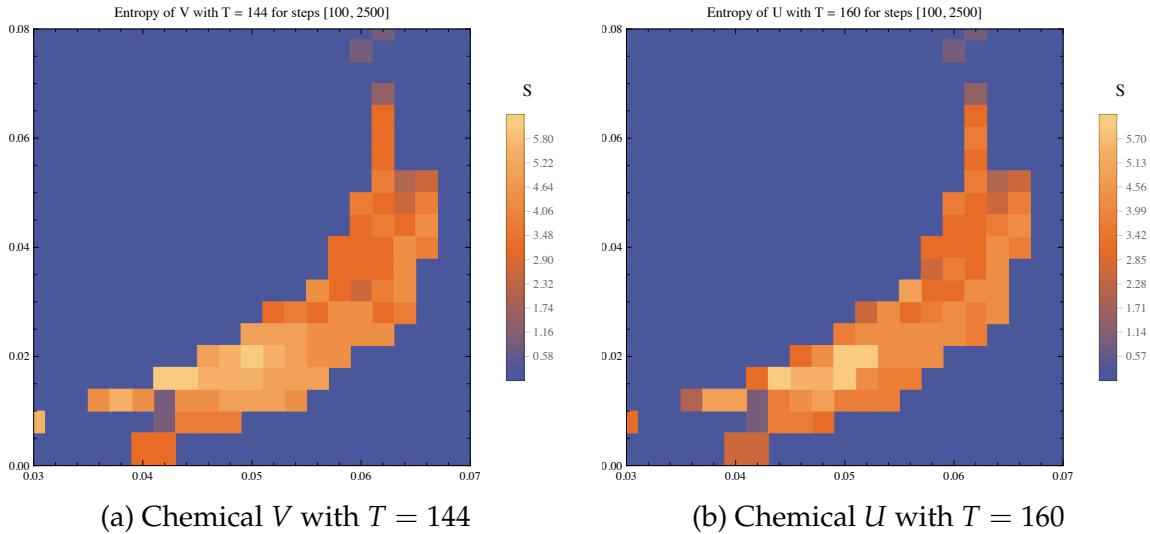


Figure 4.3: A plot of S over F, k space for both chemical V (4.3a) and U (4.3b) considering only time steps $[100, 2500]$ to remove the possible effects of initial transient states. The entropy for each chemical is slightly higher than that shown in Figure 4.2 where the initial 100 time steps are considered in the entropy calculation.

the calculation of S . We see that, in general, S is higher when the transient states are not considered. This makes sense due to the fact that there are not only fewer states to begin with but because the system is not likely to repeat a transient state. In other words, P_i is very low for $i \in [0, 100]$ and the reduction in N (a lower N in (3.2)) serves only to increase S . Either way, the calculation of S is not dramatically affected by removing the transient states.

4.3 Domain size

The domain size of the simulation can have a great effect on the entropy of the system. To illustrate this, simulations of patterns γ , ϵ , β , and δ are run with domain sizes $n = 256$, 512 , and 1024 , the results of which are shown in Figure 4.4. We see that the entropy always increases with n . This is simply due to the fact that there is more space which greatly increases the complexity of the pattern. The entropy grows faster for some patterns as n increases. **Some of the data here is fake. I do not yet have real β or δ entropy for $n = 512, 1024$. I will comment on the relative rate for growth in S for each pattern as n increases.** Figure 4.5 illustrates how varying the domain size affects the observed pattern. It is important to note that although the domain size is larger, the underlying dynamics of the system remain unchanged [18].

4.4 Other systems

One of the virtues of computational homology is its flexibility in analyzing images. To showcase this, we can examine the homology of a YouTube video. Figure 4.6 shows three frames from a YouTube video of a simulation of the SmoothLife automaton², a continuous spatial generalization of Conway’s “Game of Life” [27]. By extracting frames from the video in the form of PNG files and thresholding at $T = 128$, we can calculate the time series of Betti numbers and from there compute the entropy. We use N to indicate the total number of time steps after choosing every dt frames. There are a total of 5400 frames in the video, so the frame step size $dt = 4$, for example, gives $N = 1351$ time steps when considering the entire 3m36s video length. Table 4.1 gives the entropy S for $dt = 3$, $dt = 4$, and $dt = 5$. The entropy is highest for $dt = 3$, $S = 6.82150758771$, which is also greater than

²As of writing, the video is available at <https://youtu.be/KJe9H6qS82I>.

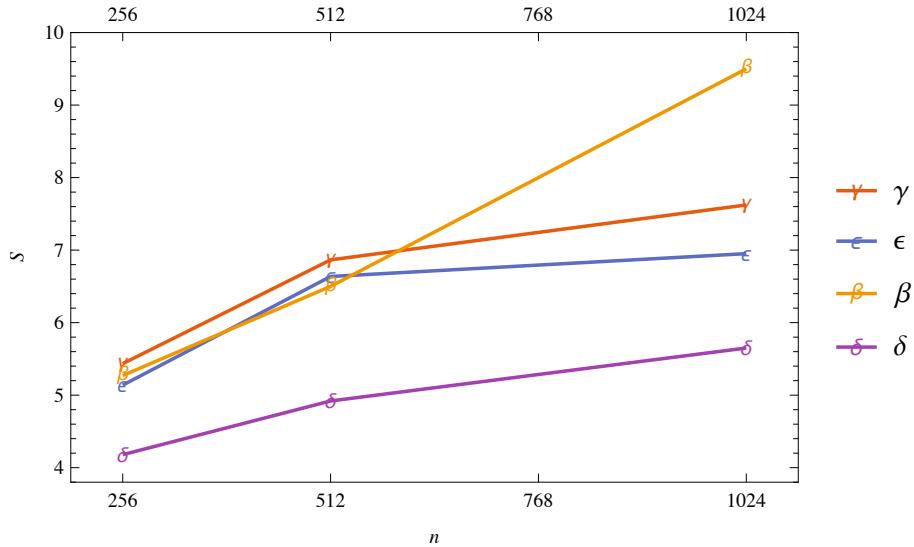


Figure 4.4: Entropy S of patterns γ , ϵ , β , and δ as the domain size n is increased. Intuitively, the entropy increases with the domain size since there is more space and therefore greater complexity. We notice that S grows faster for some patterns as n is scaled. Figure 4.5 shows pattern γ at the 2000th time step for each domain size n .

Table 4.1: The entropy S of the SmoothLife simulation video for different step sizes and number of time steps.

dt, N	3, 1800	4, 1351	5, 1081
S	6.82150758771	6.67607248	6.54613178

any S in the Gray-Scott entropy map in Figure 4.1. This is unsurprising since the SmoothLife simulation video is extremely dynamic.

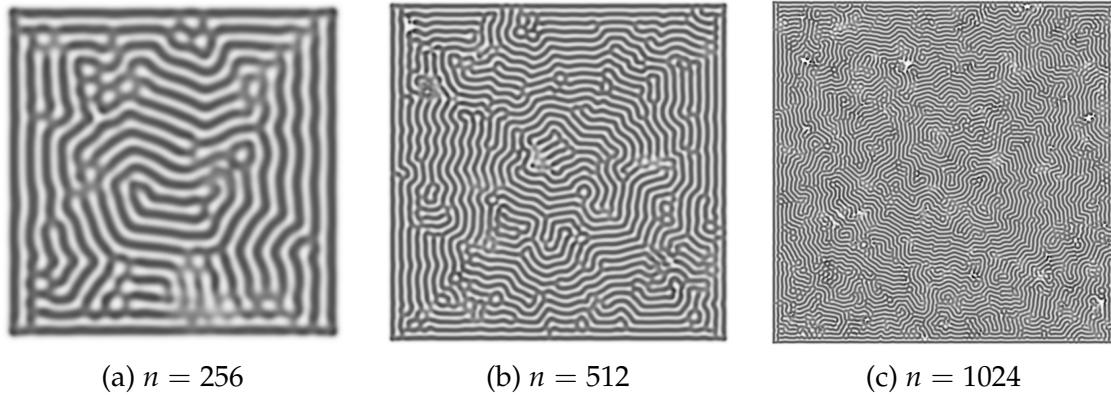


Figure 4.5: The pattern γ at the 2000th time step as the domain size is varied. The spatial complexity increases greatly as n grows. A plot of the entropy S for each $n = 256, 512, 1024$ is given in Figure 4.4.

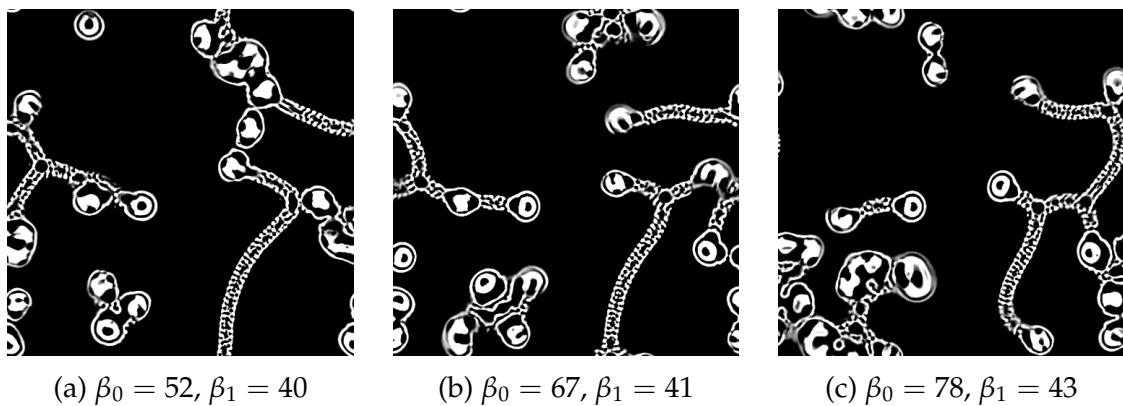


Figure 4.6: Three near-consecutive frames from a video of a SmoothLife simulation (before thresholding). The Betti numbers β_0, β_1 are given for reference. In this case, β_0 and β_1 consider white and black respectively. Computing the Betti numbers of every 3 frames gives $S = 6.82150758771$ which is highly entropic. Images adapted from S. Rafler, “Generalization of Conway’s “Game of Life” to a continuous domain—SmoothLife,” *arXiv preprint arXiv:1111.1567*, 2011. Video available at <https://youtu.be/KJe9H6qS82I>.

Conclusion

Throughout this thesis, we have tread the territory of nonlinear dynamics, computation, topology, and combinatorics. We have shown how the culmination of these theories can produce novel and compelling results when presented with complex information. By examining patterns at their lowest level, looking closely at the fundamental geometric structures that make up the dazzling images we see, we can extract meaningful information and elucidate their properties. In this case, the analysis of the Gray-Scott system led us to calculate the entropy of the system over a wide range of parameters which both complements and confirms the analysis derived directly from the physics of the system.

The homology theory presented here has given us interesting insight into the dynamics of at least one pattern-forming model, but as I have endeavored to show, one can extend these techniques to *any* topological object. This may be a single image, a video of an experiment, or a 4D construction of medical imaging data; the tools of homology are amazingly resilient and as computational methods evolve, these techniques may take precedence in the field of image analysis.

Of course, the findings in this thesis point towards many more avenues for further investigation of homology. One of the major problems confronted in this thesis is finding an appropriate threshold for which to perform the Betti number calculations. This could be solved with the implementation of an adaptive thresholding algorithm or, with a large leap in complexity, applying persistent homology (a relatively young theory at this time). Another interesting extension would be to use the time series information to derive more telling mathematical quantities such as the Lyapunov exponent which would confirm the chaotic dynamics of a system. It is certain that the applications of homology theory have not yet been exhausted; one of the virtues is that homology is so fundamental, its applications are wide open. It is my hope that the reader is convinced of its usefulness in the wake of increasingly complex data.

Appendix A

Supplemental Math

A.1 Extra definitions and theorems

Definition A.1.1. The *free abelian group generated by a finite set*

$$S = \{s_1, s_2, \dots, s_n\}$$

is the set of all functions $f : S \rightarrow \mathbb{Z}$, with the pointwise addition

$$(f + g)(s_i) := f(s_i) + g(s_i), \quad i = 1, 2, \dots, n.$$

Definition A.1.2. The *free abelian group generated by a possibly infinite set S* is the subgroup of \mathbb{Z}^S , consisting of all functions $f : S \rightarrow \mathbb{Z}$ satisfying

$$f(s) = 0 \quad \text{for all but finitely many } s \in S.$$

Appendix B

Code

B.1 Gray-Scott simulation code

The following code is written in the Python language. It requires the Numpy package which allows for easy and intuitive manipulation of matrices as well as Matplotlib to display the simulation. The procedure `RUNGS` takes four arguments: d_u , d_v , F , and k . By default, the domain size n is set to 256.

```
import numpy as np
import matplotlib.pyplot as plt

def runGS(Du, Dv, F, k):
    n = 256

    # create a structured n+2 by n+2 array of double precision floats
    Z = np.zeros((n+2,n+2), [(('U', np.double), ('V', np.double))])
    U,V = Z['U'], Z['V']

    # u, v represent the concentrations of U, V
    u,v = U[1:-1,1:-1], V[1:-1,1:-1]

    # set initial conditions
    r = 20
    u[...] = 1.0 # set all u to 1.0
    U[n/2-r:n/2+r,n/2-r:n/2+r] = 0.50
    V[n/2-r:n/2+r,n/2-r:n/2+r] = 0.25

    # 'sprinkling' of random noise
    u += 0.05*np.random.random((n,n))
    v += 0.05*np.random.random((n,n))
```

```
# set up plot
plt.ion()

# plot options
size = np.array(Z.shape)
dpi = 120.0
figsize= size[1]/float(dpi),size[0]/float(dpi)
fig = plt.figure(figsize=figsize, dpi=dpi, facecolor="white")
fig.add_axes([0.0, 0.0, 1.0, 1.0], frameon=False)
cmap = plt.cm.binary # this is a greyscale colormap
im = plt.imshow(V, interpolation='bicubic', cmap=cmap) # show V in the plot
plt.xticks([]), plt.yticks([])

# run simulation for 25000 time steps
for i in xrange(25000):
    # discretized Laplacian matrix for u
    Lu = ( U[0:-2,1:-1] + U[2: ,1:-1] +
           U[1:-1,0:-2] + U[1:-1,2: ] - 4*U[1:-1,1:-1] )

    # discretized Laplacian matrix for v
    Lv = ( V[0:-2,1:-1] + V[2: ,1:-1] +
           V[1:-1,0:-2] + V[1:-1,2: ] - 4*V[1:-1,1:-1] )

    uvv = u*v*v # the nonlinear term uv^2

    # change the concentrations in place
    u += (Du*Lu - uvv + F * (1-u))
    v += (Dv*Lv + uvv - (F+k)*v      )

    if i % 10 == 0: # show only every 10 steps on the plot
        im.set_data(V)
        im.set_clim(vmin=0.0, vmax=0.4) # set color limits
        plt.draw()
        # to save each figure
        plt.savefig('./gs/gs-%04d.png' % (i/10) ,dpi=dpi)

plt.ioff()
plt.close()
```

B.2 Entropy calculation code

The code below is used to calculate the entropy maps shown in Figures 4.1 and 4.3. It is also written in the Python language and relies on a few extra packages. The procedure BETTILIST by examining a pairs of Betti numbers contained in a CSV file with rows of the form time-step,b0,b1 where b0,b1 indicate β_0 and β_1 . MAKEP_I uses this list to count the number of times each pair {b0,b1} (interpreted simply as a string) occurs in the CSV (like a histogram). This list is normalized by the total number of pairs (equal to the number of time steps) to give a list P_i for each pair. The procedure ENTROPY takes the list of P_i and returns the entropy for the system, S .

```

import os
import subprocess
import numpy as np
from collections import Counter
import csv

# calculates entropy given a list of P_i
def entropy(P_i):
    S = 0.0
    for i in P_i:
        S += -(i*np.log(i)) # the natural log
    return S

# creates a list of all Betti number pairs {b0, b1} given a CSV file
def bettiList(csvfile):
    betti = []
    with open(csvfile, 'rU') as file:
        reader = csv.reader(file, delimiter=',')
        for row in reader:
            b0b1 = row[1: 3] # convert the b0, b1 part to a string like '[b0 b1]'
            b0b1str = ','.join(b0b1)
            betti.append(b0b1str)
    file.close()
    return betti

# makes a list of the probability of each state, P_i
def makeP_i(csvfile):
    betti = bettiList(csvfile)
    N = len(betti)
    hist = Counter(betti).items() # counts how many times each state occurs

```

```
hist.sort(lambda x, y: cmp(x[1], y[1]), reverse=True) # largest P first
# normalize
P_i = np.array([np.divide(pair[1], N, dtype=np.float) for pair in hist])
return P_i, hist, N

# saveEntropyCSV creates a CSV of entropies for each F, k pair
# input is a folder of CSVs (one for each F, k pair)
# containing {b0, b1} at each time step
def saveEntropyCSV(infolder, outfile):
    filelist = os.listdir(infolder)
    subprocess.call(['touch',outfile])
    csv = open(outfile, 'r+')
    for csvfile in filelist:
        P_i = makeP_i( infolder + '/' + csvfile )[0]
        S = entropy(P_i)
        F = csvfile.split('_')[0] # e.g. '0.044_0.038.csv'
        k = csvfile.split('_')[1][:-4] # remove '.csv'
        csv.write( F + ',' + k + ',' + str(S) + '\n')
    csv.close()
```

References

- [1] M. Cross and H. Greenside, *Pattern formation and dynamics in nonequilibrium systems*. Cambridge University Press, 2009.
- [2] K. Sims, “Reaction-diffusion tutorial,” 2013. Available at <http://www.karlsims.com/rd.html>.
- [3] H. Kurtuldu, K. Mischaikow, and M. Schatz, “Measuring the departures from the Boussinesq approximation in Rayleigh-Bénard convection experiments,” *Journal of Fluid Mechanics*, vol. 682, pp. 543–557, 2011.
- [4] S. Hui and S. Źak, “Discrete Fourier transform based pattern classifiers,” *Bulletin of the Polish Academy of Sciences: Technical Sciences*, vol. 62, no. 1, pp. 15–22, 2014.
- [5] T. Kaczynski, K. M. Mischaikow, and M. Mrozek, *Computational Homology*, vol. 157. Springer Science & Business Media, 2004.
- [6] M. Niethammer, A. N. Stein, W. D. Kalies, P. Pilarczyk, K. Mischaikow, and A. Tannenbaum, “Analysis of blood vessel topology by cubical homology,” in *Proceedings of International Conference Image Processing*, vol. 2, pp. 969–972.
- [7] M. Gameiro, K. Mischaikow, and W. Kalies, “Topological characterization of spatial-temporal chaos,” *Physical Review E*, vol. 70, no. 3, p. 035203, 2004.
- [8] A. Szymczak, A. Stillman, A. Tannenbaum, and K. Mischaikow, “Coronary vessel trees from 3d imagery: a topological approach,” *Medical Image Analysis*, vol. 10, no. 4, pp. 548–559, 2006.
- [9] A. M. Turing, “The chemical basis of morphogenesis,” *Philosophical Transactions of the Royal Society of London. Series B, Biological Sciences*, vol. 237, no. 641, pp. 37–72, 1952.

- [10] J. Bard and I. Lauder, "How well does Turing's theory of morphogenesis work?," *Journal of Theoretical Biology*, vol. 45, no. 2, pp. 501–531, 1974.
- [11] J. B. L. Bard, "A model for generating aspects of zebra and other mammalian coat patterns," *Journal of Theoretical Biology*, vol. 93, no. 2, pp. 363–385, 1981.
- [12] J. Murray, "On pattern formation mechanisms for lepidopteran wing patterns and mammalian coat markings," *Philosophical Transactions of the Royal Society B: Biological Sciences*, vol. 295, no. 1078, pp. 473–496, 1981.
- [13] H. Meinhardt, *Models of biological pattern formation*, vol. 6. Academic Press London, 1982.
- [14] E. Dabelsteen, K. Buschard, S.-I. Hakomori, and W. W. Young, "Pattern of distribution of blood group antigens on human epidermal cells during maturation," *Journal of Investigative Dermatology*, vol. 82, no. 1, pp. 13–17, 1984.
- [15] G. Turk, "Generating textures on arbitrary surfaces using reaction-diffusion," *SIGGRAPH Computer Graphics*, vol. 25, no. 4, pp. 289–298, 1991.
- [16] A. Witkin and M. Kass, "Reaction-diffusion textures," *SIGGRAPH Computer Graphics*, vol. 25, no. 4, pp. 299–308, 1991.
- [17] P. Gray and S. Scott, "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*, vol. 39, no. 6, pp. 1087–1097, 1984.
- [18] J. E. Pearson, "Complex patterns in a simple system," *Science*, vol. 261, no. 5118, pp. 189–192, 1993.
- [19] S. Strogatz, *Nonlinear Dynamics and Chaos: With Applications to Physics, Biology, Chemistry and Engineering*, vol. 272. Westview Press, 2001.
- [20] K. Krishan, H. Kurtuldu, M. F. Schatz, M. Gameiro, K. Mischaikow, and S. Madruga, "Homology and symmetry breaking in Rayleigh-Bénard convection: Experiments and simulations," *Physics of Fluids*, vol. 19, no. 11, p. 117105, 2007.
- [21] M. F. Gameiro, *Topological analysis of patterns*. Thesis, 2005.
- [22] 2Chainz, "2 Chainz (hairweavemaker)," 2015. Available at <http://instagram.com/hairweavemaker/>.

- [23] S. Harker, “Chomp,” 2015. Available at <http://chomp.rutgers.edu/>.
- [24] H. Edelsbrunner and J. Harer, “Persistent homology—A survey,” *Contemporary Mathematics*, vol. 453, pp. 257–282, 2008.
- [25] S. Weinberger, “What is... persistent homology?,” *Notices AMS*, vol. 58, no. 01, pp. 36–39, 2011.
- [26] P. Grunwald and P. Vitnyi, “Shannon information and kolmogorov complexity,” *arXiv preprint cs/0410002*, 2004.
- [27] S. Rafler, “Generalization of Conway’s “Game of Life” to a continuous domain—SmoothLife,” *arXiv preprint arXiv:1111.1567*, 2011. Video available at <https://youtu.be/KJe9H6qS82I>.