M-Lattice: A System For Signal Synthesis And Processing Based On Reaction-Diffusion

by

Alexander Semyon Sherstinsky

Master of Science
Department of Electrical Engineering and Computer Science
Massachusetts Institute of Technology
May 1989

Bachelor of Science
Department of Electrical Engineering and Computer Science
University of California Berkeley
December 1985

Submitted to the
Department of Electrical Engineering and Computer Science
in partial fulfillment of the requirements
for the degree of
Doctor of Science
at the
Massachusetts Institute of Technology
May 1994

Signature of Author ______________________________
Department of Electrical Engineering and Computer Science, 17 May 1994

Certified by ______________________________
Professor Rosalind W. Picard – Thesis Supervisor

Accepted by ______________________________
Professor Frederic R. Morgenthaler – Chairman, Departmental Committee on Theses

M-Lattice: A System For Signal Synthesis And Processing Based On Reaction-Diffusion

by

Alexander Semyon Sherstinsky

Submitted to the
Department of Electrical Engineering and Computer Science
on 17 May 1994 in partial fulfillment of the requirements
for the degree of Doctor of Science

Abstract

This research begins with reaction-diffusion, first proposed by Alan Turing in 1952 to account for morphogenesis – the formation of hydranth tentacles, leopard spots, zebra stripes, *etc.* Reaction-diffusion systems have been researched primarily by biologists working on theories of natural pattern formation and by chemists modeling dynamics of oscillating reactions. The past few years have seen a new interest in reaction-diffusion spring up within the computer graphics and image processing communities. However, reaction-diffusion systems are generally unbounded, making them impractical for many applications. In this thesis we introduce a bounded and more flexible non-linear system, the "M-lattice", which preserves the natural pattern-formation properties of reaction-diffusion.

On the theoretical front, we establish relationships between reaction-diffusion systems and paradigms in linear systems theory and certain types of artificial "neurally-inspired" systems. The M-lattice is closely related to the analog Hopfield network and the cellular neural network, but has more flexibility in how its variables interact. The bounded M-lattice enables computer or analog VLSI implementations to serve as simulation "engines" for a wide variety of systems of partial and ordinary differential equations.

On the practical front, we have developed new applications of reaction-diffusion (formulated as the new M-lattice). These include the synthesis of visual and sound textures, restoration and enhancement of fingerprints, non-linear programming, and digital halftoning of images. Halftones were synthesized in the creatively hand-drawn "special-effects" style of the Wall Street Journal portraits as well as in the "faithful-rendition" style of error-diffusion.

Thesis Supervisor: Professor Rosalind W. Picard
Title: Assistant Professor of Media Arts and Sciences,
NEC Career Development Professor Of Computers and Communication

Acknowledgements

First and foremost, I would like to acknowledge the continuous encouragement, guidance, and support I have received from my thesis supervisor, Professor Rosalind W. Picard. It has been an honor and a privilege for me to work for Roz as a research assistant. I would like to express my gratitude to her for being closely involved in all aspects of thesis research: from technical inspiration, to advice on academic and non-academic matters, to sponsoring conference trips at exotic locations, to ardent reading of electronic mail, papers, and this document. Her emphasis on clarity, depth, and rigor as well as on positive attitude, patience, caring, and good ethics has been instrumental in my own development as a scientist and as a human being. One cannot say enough good things about Roz, and I am proud to be her first doctoral graduate.

I am thankful to my thesis committee members, all of whom helped steer this research ship to the commencement harbor: Professor Berthold K. P. Horn for suggesting novel ideas, Professor George C. Verghese for asking the right questions, and Professor Gill Pratt for coming aboard on a short notice and reading the manuscript.

I have had the unique privilege to do my dissertation at the MIT Media Laboratory. Ever since joining the Vision and Modeling Group (now a part of the Perceptual Computing Section) in the Fall of 1991, I have gained tremendously from associating with its past and present inhabitants. Students and colleagues: the quadrature-mirrored Eero Simoncelli, the transparently-motion-segmented Trevor

Darrell, the steerably-filtered Bill Freeman, the motion-compensated layer-coded John Wang, the face-recognizing Martin Bichsel, the structured-from-motion Ali Azarbayejani, the finally-synchronized Martin Friedmann, fractal-based Bradley Horowitz, the orientation-sensitive Monika Gorkani, the facially-expressive Irfan Essa, the modally-matched outstanding Stan Sclaroff, the uncertainty-principled comrade George Chou, the dechirped Steve Mann, the many faces of Thad Starner, the image-compressed Bobby Desai, the promethean Chris Perry, the football-player-tracking Stephen Intille, the semantically-informed Kris Popat, the new-Wold-ordered Fang Liu, the bio-computational Baback Moghaddam, the gait-recognizing Sourabh Niyogi, the hyper-acute Yair Weiss, the ballet-dancing Lee Campbell, the gesturing Andy Wilson, the cooking Claudio Pinhanez. The post-doctoral fellows: Constance Royden, Nassir Navab. The undergraduate researchers: Tanweer Kabir, Olorunfunmi Adeyemi Oliyide, David Yu. Big thanks to the X-windowed connection-machining Thomas Minka for wonderful software. The fantastic fun-loving staff: Bea Bailey and her dog Molly, Laurie Pillsbury, Laureen Fletcher Chapman, Judy Bornstein, Jan Matlis. And, of course, thanks to the other visionaries: Professor Ted Adelson, Professor Aaron Bobick, and Professor Sandy Pentland, for their illuminating thoughts. You cannot imagine how much I have learned from all of you!

My gratitude goes to Professor Tod Machover for teaching me computer music and to to Professor V. Michael Bove Jr. for teaching me perceptual reasoning.

Thanks for expert help to the diverse contingency of other Media Lab magicians, represented here by: Dan Ellis, Michael Casey, Gilberte Houbart, Mark Lucente, Diana Dabby, Michael Hawley, Cris Dolan, Roger Kermode, Lisa Stifelman.

I have spent three years, literally day and night, at the Media Lab, a superb environment for someone who enjoys science and aesthetics. I could easily see myself being here my entire life. Thanks, for having me as a student, Media Lab!

I have been fortunate to work in two other research laboratories at MIT, prior to coming to the Media Lab: the Microsystems Technologies Laboratory and the Laboratory for Computer Science. Thanks to all the faculty, students, and staff from these research centers for helpful discussions and for social events.

I wish to give special thanks to my curriculum advisor, Professor Donald E. Troxel, for his patience, understanding, and guidance, and for serving on my Oral Qualifying Exam and on my Area Exam. His being there and willingness to share his immense knowledge and experience when I needed academic counseling has always provided needed repairs throughout my stormy eight years at MIT.

Many thanks to the many MIT faculty members, too numerous to mention here, who have contributed academically to my graduate education.

I would also like to acknowledge the financial support that I have received, in the form of research assistantship from my master's thesis supervisor, Professor Charles G. Sodini, and in the form of teaching assistantships from Professor Srinivas Devadas and from Professor Jacob White.

Others I am indebted to include Marilyn Pierce, Peggy Carney, Lisa and Rose Bella, Alice Twohig, Carolyn Zaccaria, Linda Peterson, Santina Tonelli, Benjamin Lowengard, Greg Tucker.

Thanks to all the invisible yet real persons around the globe on the multi-cultural interactive virtual-reality multi-media information super-highway for keeping me company at night.

My glorious years at MIT did not just amount to school. Special thanks goes to those who have made my extra-curricular activities enormously enjoyable as significant parts of my life. My heart is with the Russian House at MIT of the past, the present, and the future for quotes, anecdotes and jokes, creative dinners, cook-outs, and parties, and for general craziness. Thanks, Russkiy Dom, for the wonderful six years as a graduate student resident / tutor! You have turned me into a diplomat!

СПАСИБО, РУССКИЙ ДОМ !!!

Thanks to the other New House tutors, Jack and Lisa Prior, Tim and Dana Davis, and others, and to the housemater, Professer Derek Rowell, for being such a great support system.

Thanks to Bob Scanlan, Bill Fregosi, Ed Darna, and Professor Alan Brody of the MIT Music and Theater Arts Department for teaching me acting as well as other aspects of theater. Thanks to all the student members of the DRAMASHOP!

Education in general, but especially doctoral research, is associated with many years of performing creative but unstructured and stressful work on an erratic schedule. In fact, the last two weeks have been particularly strenuous. I would be unable to handle this physically-demanding feat without Joe Quinn, Sr., the head coach of the MIT ice hockey team. Thanks, coach, for keeping me on the team for seven years, and for keeping me in top shape with grueling workouts and exciting games. Thanks to the assistant coach Bill McBrine for the drills that helped me become a hockey player. Also, thanks to the other assistant coaches: Tom Keller, Joe Quinn, Jr., and Greg McManus. Thanks to all the team-mates. Hockey has certainly helped me learn a lot about myself.

MIT has given me the chance to experience college life. Thanks, MIT! (And thanks, Harvard, Boston University, and, last but not least, thanks Wellesley, too!)

Friends from Berkeley, Stanford, MIT, and beyond: you have been paramount in my endeavors. Here is a randomly-picked devil's dozen: Janet Cahn, Frandics Chan, Richard Chew, Brian Katzung, Michael Kharitonov, Tatiana Kholodenko, Dimitri Krut, Eugenia and Alla Margolin, Miroslav Predny, Irena Royzman, Chuck Rosenberg, Russ Tessier, Sasha Zubatov.

Most importantly, I am indebted to my parents, Maya and Sam Sherstinsky, grandparents Sonya and Boris Kogan, and the blessed memories of Rebecca and Mark Sherstinsky. Without my family's financial sacrifices, unwaivering emotional support, and unconditional love, this thesis would have remained wishful thinking.

I love you all.

The present document in its entirety was prepared by electronic means only, using LaTeX with the help of *idraw*, *matlab*, *mathematica*, and a host of various image-processing and graphic-arts programs.

This work was made possible through the generous contribution from Hewlett-Packard Research Laboratories, Palo Alto, California.

To my grandmother

Revekka Samsonovna Sherstinskaya

who wanted to see the completion of this thesis more than anyone else.

Also

to my mother,

to my father,

to my grandparents,

to my relatives and friends.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Alan Turing's 1952 paper [1], titled "The Chemical Basis of Morphogenesis" was a first attempt to provide a scientific explanation for the patterns of pigmentation in animals [1]. Many mammals have prominent coat markings. For example, zebras have stripes, giraffes have contoured patches, leopards and cheetahs have spots; the furs of many dogs and cats also display various forms of stripes and patches of different color. In addition, many tropical fish exhibit rich multicolored appearance. "Animals", a painting by Lenore Ramm, has been the emblem of this thesis (see Figure 1.1). The artist's harmonious portrayal of the animal kingdom reflects our fascination with the multitude of colorful textures occurring in nature [2].

1.1 Intuition Behind Turing's Model

Turing proposed to model nature's behavior by an interaction of chemicals that he called "morphogens". The simplest model uses two morphogens: the "activator" and the "inhibitor". The morphogens themselves are produced by chemical reactions among particular enzymes in every cell of the animal's skin during the animal's

[1] This was the British mathematician's last published work before his untimely death.

Figure 1.1: “Animals”, a painting by Lenore Ramm.

embryonic stages.

According to this model, the two morphogens react with each other; however, the model consisting of reaction alone cannot account for the tremendous variety of coating patterns observed in animals. Since there is no inter-cellular flow of morphogens in the model, every cell acts as an independent autonomous system, producing the final morphogen concentrations based only on random initial concentrations. Therefore, cells end up in stable states that have no correlation or spatial structure, unlike the majority of patterns occurring in nature. In order to supplement the model with the needed transport mechanism, Turing incorporated a diffusion term into the system of equations. Then he showed mathematically that this reaction-diffusion system is capable of producing a wide variety of spatial patterns.

To gain a qualitative understanding of the operation of a two-morphogen reaction-diffusion system, consider two morphogens, the activator and the inhibitor, each reacting with itself and the other. While the reactions influence local concentrations of the two morphogens, the diffusion transports the morphogens from cell to cell. These interactions are depicted in Figure 1.2. Suppose the activator is auto-catalytic but diffuses slowly. In other words, its concentration increases in proportion to the amount already present, but its diffusion rate is low compared to that of the inhibitor. Thus the activator and the inhibitor create two opposing tendencies. On one hand, the activator concentration grows at a high rate locally, but does not spread fast enough to replace the inhibitor everywhere. On the other hand, the inhibitor consumes the activator at a low rate locally, but, because of its high diffusion constant, the inhibitor is delivered faster to remote sites, keeping the activator concentration finite everywhere. The competition between these two tendencies causes the concentration profiles of the activator and the inhibitor to settle into patterns of peaks and valleys, 180° out of phase with each other [2] [3].

[2]Precise explanation follows in Chapter 3.

$$\frac{\partial \Psi_A(x, t)}{\partial t} = \Psi_A(x, t)\, \Psi_I(x, t) - \Psi_A(x, t) - 12 + D_A \frac{\partial^2 \Psi_A(x, t)}{\partial x^2}$$

$$\frac{\partial \Psi_I(x, t)}{\partial t} = 16 - \Psi_A(x, t)\, \Psi_I(x, t) + D_I \frac{\partial^2 \Psi_I(x, t)}{\partial x^2}$$

$\Psi_A(x, t)$: activator concentration

$\Psi_I(x, t)$: inhibitor concentration

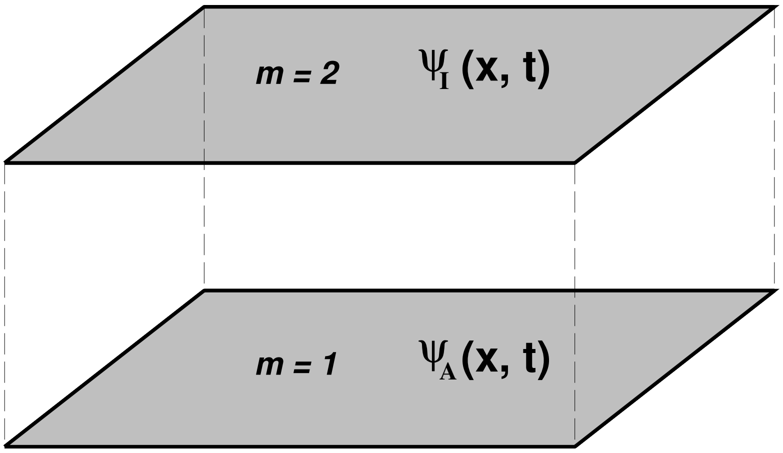


Figure 1.2: Turing's original two-morphogen reaction-diffusion model. Interactions between the activator and the inhibitor morphogens are shown on a 2-D spatial lattice. Each morphogen diffuses horizontally on its separate layer and reacts vertically across layers.

1.2 Testing Turing's System On Hydranth

As a case study, Turing modeled the tentacle formation in hydranth (a small tubular fresh-water polyp, shown in Figure 1.3) with a 1-D reaction-diffusion system. Since high-speed digital computers were unavailable in his time, Turing was unable to perform extensive computer simulations of various 1-D and 2-D reaction-diffusion systems. His approach employed the linear analysis of the 1-D non-linear system for small deviations of morphogen concentrations around the equilibrium point. He then argued that the 1-D system's non-linear behavior, *i.e.*, its behavior for large excursions of morphogen concentrations away from the equilibrium point, is not significantly different qualitatively from the linearized behavior. For the 2-D case, making such an argument was and still is impossible.

The simple 1-D reaction-diffusion model exhibited several properties desirable from the biological point of view. For instance, the resulting spatial patterns display a characteristic wavelength, called the "chemical wavelength", which manifests itself as a periodic repetition of white and black segments on the hydranth. The chemical wavelength depends strongly on the relative reaction and diffusion rates of the two morphogens but only weakly on the initial morphogen concentrations. This means that all hydranths look similar but every individual hydranth is slightly different.

Another interesting property is that the time it takes for the model to produce a stationary pattern [3] is comparable to the time it takes for such a pattern to form in an actual embryo. Incidentally, this property has been the subject of debate in the theoretical biology literature. Much later, in 1974, Bard and Lauder conducted a number of computer experiments with the 1-D and 2-D Turing reaction-diffusion

[3] This is referring not to the real time it takes for a human or a computer to complete a simulation of the reaction-diffusion system and plot the morphogen concentrations, but to the absolute time constant (in seconds), which is a property of the reaction-diffusion system.

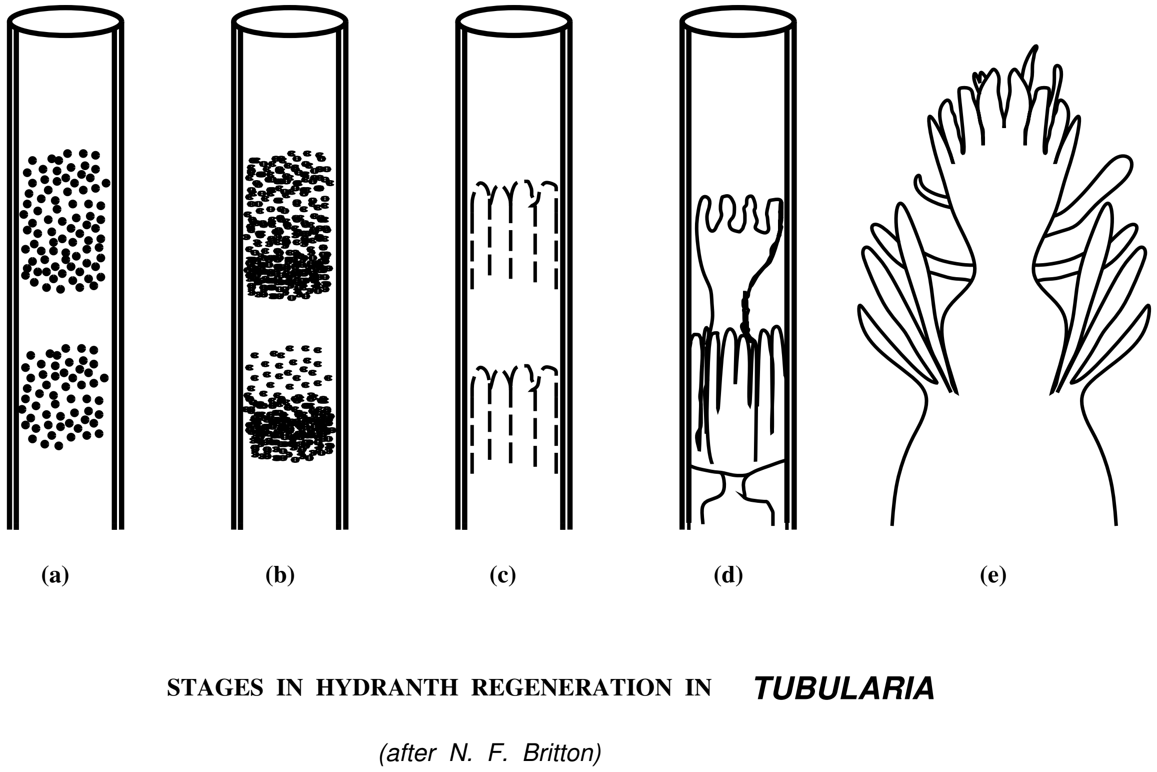


Figure 1.3: The application of Turing's original reaction-diffusion model to the hydranth, a 1-D organism. The chemical wavelength, predicted by the 1-D linear analysis, is manifested by a periodic spacing of tentacles in the tubularia.

models in order to test them on biological data [4].

1.3 General Morphogenesis Mechanism

Arguably, one of the most important properties of reaction-diffusion systems is the commonality of their constituent processes, reaction and diffusion. It is conjectured that a simple and flexible mechanism is more feasible than a complex and rare one from the evolutionary point of view. Probabilistically, nature is more likely to adopt a process that requires only a few steps and ingredients than a process that depends on many conditions that are rarely met together. The simplicity of a chemical reaction among the readily available enzymes coupled with such a basic and ubiquitous natural process as diffusion has won the reaction-diffusion model a serious consideration as a physically-based model for natural pattern formation. All together, these three properties have made the reaction-diffusion phenomenon a realistic candidate for modeling the process of morphogenesis [5].

The general belief that no single reaction-diffusion model is adequate in all cases has led researchers to many variants and extensions of Turing's basic two-morphogen system [6], [7]. Altering the form of reaction among morphogens and / or varying the number of morphogens in the system produces different concentration profiles. For example, a five-morphogen model due to Meinhardt has been found to agree with experimental data collected from various biological systems, such as fruit flies and sea shells. An extensive study of parameters for this model has been done by Murray [8], [9], [10]. To complete the list of classical reaction-diffusion publications, a broad mathematical overview of various reaction-diffusion systems, fortified with an exhaustive listing of references, can be found in [11], [12].

1.4 Spread Of Interest In Reaction-Diffusion

The fundamental reaction-diffusion system, comprised by (possibly non-linear) partial differential equations (PDEs) in several spatial and one temporal dimensions is very general, but its various aspects and capabilities can be emphasized by the choice of reaction functions and model parameters [4].

For nearly four decades, the reaction-diffusion systems stayed primarily in the

[4]For instance, under a certain choice of the reaction function and the diffusion constant, even the Shröedinger equation can be written as a reaction-diffusion system.

domain of theoretical biologists working on theories of morphogenesis, until Murray's 1988 paper in *Scientific American* sparked a wide-spread interest in reaction-diffusion systems [13]. In particular, since 1990, there has been a significant surge in interest in reaction-diffusion systems within the computer graphics and image processing communities. The graphics community's interest in reaction-diffusion systems stems from their ability to model and synthesize directly on a three-dimensional object a wide class of natural textures. The various animal coating patterns that Turing's theory set out to model can now be generated directly on 3-D surfaces of any shape via finite difference methods. The speed of simulating reaction-diffusion systems on a digital computer and the mapping of rectangular 2-D textures to 3-D surfaces still remain as challenging issues [14], [15]. In the realm of image processing, Price used Meinhardt's five-morphogen reaction-diffusion system for texture classification and for fingerprint matching [16]. However, a formal justification for using reaction-diffusion systems in image processing is still lacking.

The large amount of attention reaction-diffusion systems are receiving from researchers in the engineering disciplines can also be attributed to their inherently non-linear nature [17], [18]. While linear systems have been the prevalent engineering tool, the improvement in performance that results from refining the linear model for many applications has diminished, and the actual performance has saturated. In contrast, non-linear systems are poorly understood, but understanding them might help overcome the limitations of linear models in certain problems. Depending on the performance criteria, the payoff brought about by employing a non-linear model may be significantly greater than that achievable with a linear model. For example, some recently published papers in the image processing and pattern recognition literature have explored the application of non-linear PDEs to texture classification and contour detection [16], [19], [20].

1.5 Practical Difficulties

In order to possess pattern formation properties in the sense of Turing, a reaction-diffusion system must exhibit local instability to small non-homogeneous perturbations. In addition, practical considerations dictate that the system should be bounded in the large-signal sense. A major difficulty associated with the reaction-diffusion system paradigm in its standard form is that the system is totally stable, or even bounded, only for a restricted class of non-linear reaction functions. This drawback does not present a problem for actual biological systems, because in nature, morphogen concentrations cannot be negative, nor can they be too large without depleting the supply. On the contrary, it does narrow the scope of the model's engineering applications.

A common approach aimed at preventing numerical overflow from plaguing the simulations of reaction-diffusion systems on the digital computer has been to clip the magnitudes of the state variables by adding a special clause (*e.g.*, an "if" statement) to the numerical method (*e.g.*, Forward Euler) used for solving the system of differential equations [15]. For some reaction-diffusion systems, this technique eventually manages to stop the state variables from changing between successive time steps, even when the actual state variables are supposed to have non-zero time derivatives. In general, clipping the state variables of a system of differential equations from within the numerical method destroys the mathematical integrity of the original dynamical system, thereby complicating the analysis.

1.6 Proposed Solutions And Other Contributions

The present work makes both theoretical and practical contributions. The main contribution of this research is the formulation of the M-lattice system. The clipping introduced by the M-lattice system addresses the issue of large-signal boundedness of reaction-diffusion systems in a way that does not dismiss the non-linearities. Due to its flexibility, the M-lattice system can be employed to simulate reaction-diffusion as well as many other non-linear dynamical systems, and it can still be analyzed mathematically. We have identified three main modes of operation of the M-lattice system: texture synthesis, adaptive filtering, and non-linear optimization. Thus, using the M-lattice system for signal synthesis and processing is justified by formally matching one of these modes to the specific computational task chosen for the given application.

In contrast with the original reaction-diffusion system, various degrees of stability of the M-lattice system have been observed in computer simulation for many practical non-linear reaction functions. In order to account for some of these observations, we prove the total stability of a subclass of the M-lattice system.

As part of the introduction of this general practical model to the signal processing community, we establish relationships among reaction-diffusion systems and the well-known paradigms in linear systems theory and artificial neural systems.

From the practical standpoint, we apply the M-lattice system to the synthesis of visual and sound textures, digital halftoning of images, and restoration and enhancement of fingerprints.

The rest of this document is organized as follows. Chapter 2 presents the mathematical background of typical reaction-diffusion systems. Chapter 3 introduces the linear-reaction reaction-diffusion system, generalizes it to the convolutionally-coupled M-lattice system, and examines the latter in the framework of Fourier analysis. Chapter 4 presents the mathematical treatment of the M-lattice system and the clipped M-lattice system. Chapter 5 illustrates the application of the clipped M-lattice system to digital halftoning of images and to restoration and halftoning of fingerprints. Finally, Chapter 6 summarizes the research.

Chapter 2

Background

Much of the material in this chapter appears in various literature sources. It is presented here using notation that will simplify the comparison of the reaction-diffusion system to related models and facilitate the development of the M-lattice system.

2.1 Reaction-Diffusion Formulation

Let $\psi_m(\vec{x}, t) \in \Re$ be the concentration of the m-th morphogen ($m = 1, \ldots, M$) as a function of d-dimensional (d-D) space $\vec{x} \in \Re^d$ and of time $t \in \Re_+$. Denote the vector of all morphogen concentrations by $\vec{\psi}(\vec{x}, t) \in \Re^M = [\psi_1(\vec{x}, t), \ \ldots, \ \psi_M(\vec{x}, t)]^T$. Then reactions among various morphogens are prescribed by $R_m(\vec{\psi}(\vec{x}, t))$, which is a possibly non-linear function for every morphogen. Each morphogen also undergoes steady Fickian diffusion [1], and $D_m \in \Re_+$

[1] Fick's law of diffusion says that the flux of material is proportional to the gradient of the concentration of the material.

is the diffusion constant of the m-th morphogen (the quadratic form stresses its positivity). Convective flow of any morphogen is described by velocity, $\vec{v}_m \in \Re^d$, and $b_m \in \Re$ is the dissipation constant of the m-th morphogen. Using this nomenclature, we define the reaction-diffusion system.

Definition 2.1.1 *General reaction-diffusion system equations.*

$$\frac{\partial \psi_m(\vec{x},t)}{\partial t} = D_m \nabla^2 \psi_m(\vec{x},t) - \vec{v}_m^T \nabla \psi_m(\vec{x},t) - r_m \psi_m(\vec{x},t) + R_m(\vec{\psi}(\vec{x},t)). \quad (2.1)$$

All the linear interactions in (2.1) can be combined into $\mathcal{D}_m$, a general derivative operator of an arbitrary order, which absorbs diffusion constants, velocity (of convection) vectors, dissipation rates, and any other scaling factors with appropriate units. For the M-vector of morphogens, define $\vec{\mathcal{D}} = [\mathcal{D}_1, \ \ldots, \ \mathcal{D}_M]^T$.

Reformulating (2.1) by applying the linear derivative operator, $\mathcal{D}_m$, to $\psi_m(\vec{x},t)$ gives:

$$\frac{\partial \psi_m(\vec{x},t)}{\partial t} = \mathcal{D} \psi_m(\vec{x},t) + R_m(\vec{\psi}(\vec{x},t)) \quad (2.2)$$

for each morphogen.

By using $\vec{\mathcal{D}}$ and defining $\vec{R}(\vec{\psi}(\vec{x},t)) = \left[R_1(\vec{\psi}(\vec{x},t)), \ \ldots, \ R_M(\vec{\psi}(\vec{x},t))\right]^T$, we arrive at the definition of the general reaction-diffusion system:

$$\frac{\partial \vec{\psi}(\vec{x},t)}{\partial t} = \vec{\mathcal{D}} \vec{\psi}(\vec{x},t) + \vec{R}(\vec{\psi}(\vec{x},t)). \quad (2.3)$$

2.2 Mathematical Predecessors

In what is considered as one of the most important papers in theoretical biology this century, Turing (1952) pioneered the study of this general multi-morphogen reaction-diffusion model, known as the interacting-population reaction-diffusion system in the mathematical biology literature. Although appearing concise and deceptively simple, (2.3) distills over a century of research.

The field's "official" beginning dates back to 1836, when Verhulst proposed the now familiar logistic growth model for studying population dynamics [2] for single species. A notable application of a related model is for the spruce budworm outbreak, a major problem in Canada.

Initial studies of population growth dealt with differential equations representing small subsets of (2.3). For instance, in the language of reaction-diffusion, the single-species models require only one morphogen, which automatically implies that they do not involve reaction. Likewise, these simple models do not take into account the spatial detail and spread of the population, meaning that there is no diffusion.

The first interacting-species model was proposed by Voltera (1926) to explain oscillatory levels of certain fish catches in the Adriatic. This has led to many systems aimed at studying the predation of one species by another. These equations are now commonly classified as predator-prey models. They are also known as the Lotka-Voltera systems, since the same equations were also derived by Lotka (1920) from a hypothetical chemical reaction, which could exhibit periodic behavior in chemical concentrations. In the reaction-diffusion lingo, these systems have reaction, but no diffusion.

The studies of biological oscillators and switches comprise another area of research that employs diffusionless multi-species models. The systems range from the simple two-species oscillators, such as the "Brusselator", to more complex models, such as the one resulting from the Hodgkin-Huxley (1952) theory of nerve membranes. The reduced analytically tractable version, due to FitzHugh-Nagumo (1961) contains

[2] The exponential population growth model, due to the infamous Malthus in 1798, but actually suggested earlier by the famous Euler, was rejected as unrealistic.

three morphogens.

Studies have shown that diffusion models form a reasonable basis for studying the dispersal of interacting and competing species of insects and animals. As a model for the spread of an advantageous gene in a population, Fisher (1937) augmented the logistic growth model by adding a diffusion term with a constant diffusion coefficient. The resulting equation is now known as the Fisher equation and is the first single-species (or single-morphogen, or reactionless) reaction-diffusion system.

The most widely studied, both theoretically and experimentally, oscillating chemical reaction is the Belousov-Zhabotinsky reaction (1951). The Field-Noyes (1974) model quantitatively mimics the actual chemical reactions. This system, sometimes referred to as the "Oregonator", is a three-morphogen non-linear diffusionless reaction-diffusion system. But by allowing the reactants to diffuse at a constant rate, almost all the phenomena theoretically exhibited by reaction-diffusion mechanisms have been found in this real and practical reaction.

Building upon the basic knowledge of calculus, linear algebra, and differential equations, the books by Murray [12], Britton [11], Meinhardt [6], Segel [21], and Strogatz [22] provide informative and inspiring excursions into this fascinating interdisciplinary science.

2.3 Turing's Reaction-Diffusion System

The original 1-D and 2-D reaction-diffusion systems developed by Turing and studied by Bard use two morphogens and have no convection or dissipation. Let $\psi_A(\vec{x}, t)$ and $\psi_I(\vec{x}, t)$ be the concentrations of the activator and the inhibitor, respectively. Then for Turing's reaction-diffusion model:

$$
\begin{aligned}
R_A(\vec{\psi}(\vec{x},t)) &= \psi_A(\vec{x},t)\cdot\psi_I(\vec{x},t) - \psi_A(\vec{x},t) - 12 + q(\vec{x}), \\
R_I(\vec{\psi}(\vec{x},t)) &= 16 - \psi_A(\vec{x},t)\cdot\psi_I(\vec{x},t), \\
\mathcal{D}\psi_A(\vec{x},t) &= D_A\nabla^2\psi_A(\vec{x},t), \\
\mathcal{D}\psi_I(\vec{x},t) &= D_I\nabla^2\psi_I(\vec{x},t). \qquad (2.4)
\end{aligned}
$$

where $q(\vec{x})$, called the "evocator", is a waveform of small random perturbations, and D_A and D_I are the respective diffusion rates. For the 1-D case, the Laplacian reduces to the second derivative.

The evocator is crucial to the operation of any reaction-diffusion system. To see this in the particular case of Turing's reaction-diffusion system, note that if $q(\vec{x}) = 0$ and $\psi_A(\vec{x}, t = t_0) = \psi_I(\vec{x}, t = t_0) = 4$, then $\dfrac{\partial\psi_m(\vec{x},t)}{\partial t}$ will be identically zero for both morphogens for all t. In other words, the concentrations of both morphogens in every cell are in equilibrium, and will remain in equilibrium forever. According to the reaction-diffusion model, nature causes the concentrations of various chemicals in the neighboring cells of an embryo to be slightly and randomly mismatched. This creates the evocator, which gives rise to diffusion. The diffusion, in turn, sets up a non-zero $\dfrac{\partial\psi_A(\vec{x},t)}{\partial t}$. The changing activator concentration causes the inhibitor concentration to change with time as well, and the system moves away from equilibrium.

Note that any homogeneous perturbations ($q(\vec{x}) = q_0$) to the reaction-diffusion system in equilibrium decay with time, since a homogeneous perturbation cannot give rise to diffusion. In the absence of diffusion, there is no flux of chemicals, and each cell functions as an isolated sub-system. Nature returns every cell of a homogeneously perturbed reaction-diffusion system to equilibrium. In contrast, a non-homogeneous perturbation does give rise to diffusion. The presence of diffusion is essential to the generation of non-trivial spatial patterns. Stability to homogeneous perturbations and instability to non-homogeneous perturbations is one of the key mathematical properties of reaction-diffusion systems, to which we will return numerous times throughout this document. Diffusion alone, however, cannot produce non-trivial spatial patterns. Without reaction, the system becomes a heat equation, which is known to smooth out any non-homogeneous initial condition.

2.4 Analysis Of The 1-D Turing System

This section, along with Section B.3 of Appendinx B, can serve as a tutorial on the basic analysis of complex non-linear dynamical systems. We derive the key properties of Turing's original two-morphogen reaction-diffusion system using the standard tool-kit consisting of discretization, linearization, and the Discrete Fourier Transform (DFT).

For $x \in \Re$, (2.4) becomes:

$$\begin{aligned} \frac{\partial \psi_A(x,t)}{\partial t} &= \psi_A(x,t)\cdot\psi_I(x,t) - \psi_A(x,t) - 12 + q(x) + D_A\frac{\partial^2\psi_A(x,t)}{\partial x^2}, \\ \frac{\partial \psi_I(x,t)}{\partial t} &= 16 - \psi_A(x,t)\cdot\psi_I(x,t) + D_I\frac{\partial^2\psi_I(x,t)}{\partial x^2}. \end{aligned} \tag{2.5}$$

2.4.1 Discretization And Linearization

We now assume that the cells of an animal's skin are equally spaced and comprise a periodic 1-D lattice with the period of N_x cells. Using a popular discretization of the second derivative [23], turns (2.5) into:

$$\begin{aligned} \frac{d\psi_A(n_x,t)}{dt} &= \psi_A(n_x,t)\cdot\psi_I(n_x,t) - \psi_A(n_x,t) - 12 + q(n_x) \\ &+ D_A[\psi_A(n_x+1,t) - 2\psi_A(n_x,t) + \psi_A(n_x-1,t)], \\ \frac{d\psi_I(n_x,t)}{dt} &= 16 - \psi_A(n_x,t)\cdot\psi_I(n_x,t) \\ &+ D_I[\psi_I(n_x+1,t) - 2\psi_I(n_x,t) + \psi_I(n_x-1,t)], \\ \psi_A(n_x+N_x,t) &= \psi_A(n_x,t), \\ \psi_I(n_x+N_x,t) &= \psi_I(n_x,t). \end{aligned} \tag{2.6}$$

(See Section B.3 for a more general treatment of discretization.) The objective is to solve for each morphogen concentration as a function of n_x and t. There are two difficulties: the system is non-linear in $\psi_A(n_x, t)$ and $\psi_I(n_x, t)$, and the spatial variables are intermixed. In other words, the morphogen concentration at every spatial index in (2.6) depends on the spatial convolution involving morphogen concentrations at other indices as well. A common approach used in dealing with such problems is to linearize the system for small deviations of the concentrations from the equilibrium (or critical, or fixed) point and then to separate the spatial variables with the DFT, which turns convolutions into multiplications. For future reference, the DFT and its inverse transform, IDFT, are:

$$\Psi[k_x] = \sum_{n_x=0}^{N_x-1} \exp\left\{-j\frac{2\pi}{N_x}k_x n_x\right\} \psi[n_x], \tag{2.7}$$

$$\psi[n_x] = \frac{1}{N_x}\sum_{k_x=0}^{N_x-1} \exp\left\{j\frac{2\pi}{N_x}k_x n_x\right\} \Psi[k_x]. \tag{2.8}$$

First, we approximate every function of the two morphogen concentrations with a Taylor series expansion about the equilibrium, retaining only the linear terms:

$$\begin{aligned} f[\psi_A(n_x,t), \psi_I(n_x,t)] &\approx f[\psi_{A,eq}(n_x,t), \psi_{I,eq}(n_x,t)] \\ &+ \frac{\partial[f(\psi_A(n_x,t), \psi_I(n_x,t))]}{\partial[\psi_A(n_x,t)]}\psi_{A,s}(n_x,t) \\ &+ \frac{\partial[f(\psi_A(n_x,t), \psi_I(n_x,t))]}{\partial[\psi_I(n_x,t)]}\psi_{I,s}(n_x,t), \end{aligned} \tag{2.9}$$

where the subscript "eq" denotes the equilibrium value, and the subscript "s" denotes a small deviation from the equilibrium value. Applying (2.9) to (2.6) and evaluating $\frac{\partial[f(\psi_A(n_x,t), \psi_I(n_x,t))]}{\partial[\psi_A(n_x,t)]}$ and $\frac{\partial[f(\psi_A(n_x,t), \psi_I(n_x,t))]}{\partial[\psi_I(n_x,t)]}$ at $\psi_{A,eq}(n_x, t = t_0) = \psi_{I,eq}(n_x, t = t_0) = 4$ produces the following "small-signal" reaction-diffusion system:

$$
\begin{aligned}
\frac{d\psi_{A,s}(n_x,t)}{dt} &= 3\psi_{A,s}(n_x,t) + 4\psi_{I,s}(n_x,t) \\
&+ D_A[\psi_{A,s}(n_x+1,t) - 2\psi_{A,s}(n_x,t) + \psi_{A,s}(n_x-1,t)], \\
\frac{d\psi_{I,s}(n_x,t)}{dt} &= -4\psi_{A,s}(n_x,t) - 4\psi_{I,s}(n_x,t) \\
&+ D_I[\psi_{I,s}(n_x+1,t) - 2\psi_{I,s}(n_x,t) + \psi_I(n_x-1,t)], \\
\psi_{A,s}(n_x+N_x,t) &= \psi_{A,s}(n_x,t), \\
\psi_{I,s}(n_x+N_x,t) &= \psi_{I,s}(n_x,t). \qquad (2.10)
\end{aligned}
$$

2.4.2 Separation Of Variables

Thus far, the combination of discretization and linearization has turned spatial derivatives into spatial convolutions, making the variables corresponding to different spatial wave numbers in (2.10) intermixed [3]. Variables are separated in a standard way by applying the DFT, (2.7), to every term of (2.10), turning convolutions into multiplications:

$$
\begin{aligned}
\frac{\partial\psi_{A,s}(k_x,t)}{\partial t} &= 3\psi_{A,s}(k_x,t) + 4\psi_{I,s}(k_x,t) - 4D_A\psi_{A,s}(k_x,t)\sin^2\left(\frac{\pi k_x}{N_x}\right), \\
\frac{\partial\psi_{I,s}(k_x,t)}{\partial t} &= -4\psi_{A,s}(k_x,t) - 4\psi_{I,s}(k_x,t) - 4D_I\psi_{I,s}(k_x,t)\sin^2\left(\frac{\pi k_x}{N_x}\right), \\
\psi_{A,s}(k_x+N_x,t) &= \psi_{A,s}(k_x,t), \\
\psi_{I,s}(k_x+N_x,t) &= \psi_{I,s}(k_x,t). \qquad (2.11)
\end{aligned}
$$

Since both sides of (2.11) depend on the morphogen concentrations at only one spatial wave number, the variables are now separated.

[3]This convolutional mixing of variables is not to be confused with the kind of mixing and separation of spatial and temporal variables commonly encountered in the studies of PDEs.

2.4.3 Solution Modes Of The System

The two equations comprise a linear system. Thus, it is advantageous to study a general two-morphogen linear system of the form:

$$\begin{aligned}\frac{\partial\psi_{A,s}(k_x,t)}{\partial t} &= \psi_{A,s}(k_x,t)\left[r_{11}-4D_A\sin^2\left(\frac{\pi k_x}{N_x}\right)\right]+r_{12}\psi_{I,s}(k_x,t),\\ \frac{\partial\psi_{I,s}(k_x,t)}{\partial t} &= r_{21}\psi_{A,s}(k_x,t)+\psi_{I,s}(k_x,t)\left[r_{22}-4D_I\sin^2\left(\frac{\pi k_x}{N_x}\right)\right],\end{aligned}\tag{2.12}$$

where the diffusion rates, D_A and D_I, are restricted to be non-negative. The constants $r_{m_1m_2}$ are called the *marginal reaction rates.* Expressed using the matrix notation, (2.12) becomes:

$$\begin{bmatrix}\frac{d\psi_{A,s}}{dt}\\ \frac{d\psi_{I,s}}{dt}\end{bmatrix}=\begin{bmatrix} r_{11}-4D_A\sin^2\left(\frac{\pi k_x}{N_x}\right) & r_{12}\\ r_{21} & r_{22}-4D_I\sin^2\left(\frac{\pi k_x}{N_x}\right)\end{bmatrix}\begin{bmatrix}\psi_{A,s}\\ \psi_{I,s}\end{bmatrix}\tag{2.13}$$

for each spatial wave number, k_x.

Depending on the eigenvalues and the initial conditions, the system (2.13) can exhibit six types of solutions:

- $\lambda_1(k_x)$ and $\lambda_2(k_x)$ are a complex pair with a negative real part $\Longrightarrow$ decaying traveling waves;
- $\lambda_1(k_x)$ and $\lambda_2(k_x)$ are a complex pair with a positive real part $\Longrightarrow$ growing traveling waves;
- two identical decaying traveling waves moving in the opposite directions $\Longrightarrow$ decaying standing waves;
- two identical growing traveling waves moving in the opposite directions $\Longrightarrow$ growing standing waves;

class of wave	both $\Re[\lambda] < 0$ (stable)	either $\Re[\lambda] \geq 0$ (unstable)	λ	type of solution
traveling	decaying	growing	complex	oscillatory
standing	decaying (sum of a decaying traveling wave and its reflection)	growing (sum of a growing traveling wave and its reflection)	complex	oscillatory
stationary	decaying	growing	real	non-oscillatory

Table 2.1: The kinds of modes admitted by a 2×2 reaction-diffusion system. The terms "decaying" and "growing" refer to the temporal behavior.

- both $\lambda_1(k_x)$ and $\lambda_2(k_x)$ are real and negative $\Longrightarrow$ decaying spatially-stationary waves; and
- either $\lambda(k_x)$ is real and positive $\Longrightarrow$ growing spatially-stationary waves.

The solution is a non-stationary spatial wave, unless $\lambda(k_x)$ is real. Both traveling wave and standing wave solutions are called non-stationary, because the amplitudes of such waves undergo sign changes. Table 2.1 summarizes all the possibilities for a 2×2 reaction-diffusion system.

Traveling waves cannot model an animal coat texture, because they do not produce a constant spatial pattern. Also, if the real part of $\lambda(k_x)$ is negative, then the spatial harmonics decay to zero. Thus, for explaining the formation of natural patterns, such as zebra stripes and leopard spots, the last mode has received attention. The other modes of reaction-diffusion systems have also been used, for instance, in modeling the behavior of oscillating chemical reactions [11], [12].

The only mode of the system in (2.13) that is capable of producing stationary spatial waves is the one corresponding to $\lambda(k_x) \in \Re,\ \lambda(k_x) > 0$ for some range of k_x. Since the amplitude of every k_x that belongs to this band of spatial frequencies grows as a function of time, the system becomes unstable for that particular spatial frequency. Therefore, in order to produce stationary spatial waves, the system must be unstable for at least one spatial frequency. The harmonic $k_x = 0$ is excluded from the band of unstable wave numbers by definition so as to maintain stable equilibrium levels in the absence of diffusion. Thus, the system's final output should be unaffected by homogeneous perturbations.

Since (2.13) is a second-order system, it has two eigenvalues. Clearly, as time increases, the solution will become dominated by the mode whose spatial wave number corresponds to the eigenvalue having the most positive real part. As we shall see shortly, only one eigenvalue of Turing's system can be made positive. The other eigenvalue is always negative for all spatial wave numbers. Assuming that the eigenvalues are distinct, let $\lambda(k_x)$ be the dominant eigenvalue as a function of the spatial harmonic, k_x. Since the range of $k_x = 1,\ \ldots,\ N_x$ is limited, the real part of $\lambda(k_x)$ will attain a maximum for at least one k_x^* [4]. In other words, the dominant eigenvalue will have at least one dominant mode. Hence, using the IDFT, (2.8), on (2.13) produces the formula for the activator concentration:

$$\lim_{t\to\infty} \psi_{A,s}(n_x, t) \quad = \quad \frac{1}{N_x} C(k_x^*) \exp\left\{ j \frac{2\pi}{N_x} k_x^* n_x + \lambda(k_x^*) t \right\}, \tag{2.14}$$

where $C(k_x)$ depends on the eigenvectors and the initial conditions at k_x^*. As an illustration, Figure 2.1 plots the spatial-frequency-index response of the dominant eigenvalue for a system capable of pattern formation.

We have established that (2.14) is the elementary mode of (2.13) that is relevant to pattern formation. The next task is to identify the conditions on D_A and D_I, which enable the system in (2.13) to actually become unstable, and to determine

[4] In Chapter 5, we will show that one can design the reaction-diffusion system to have such properties.

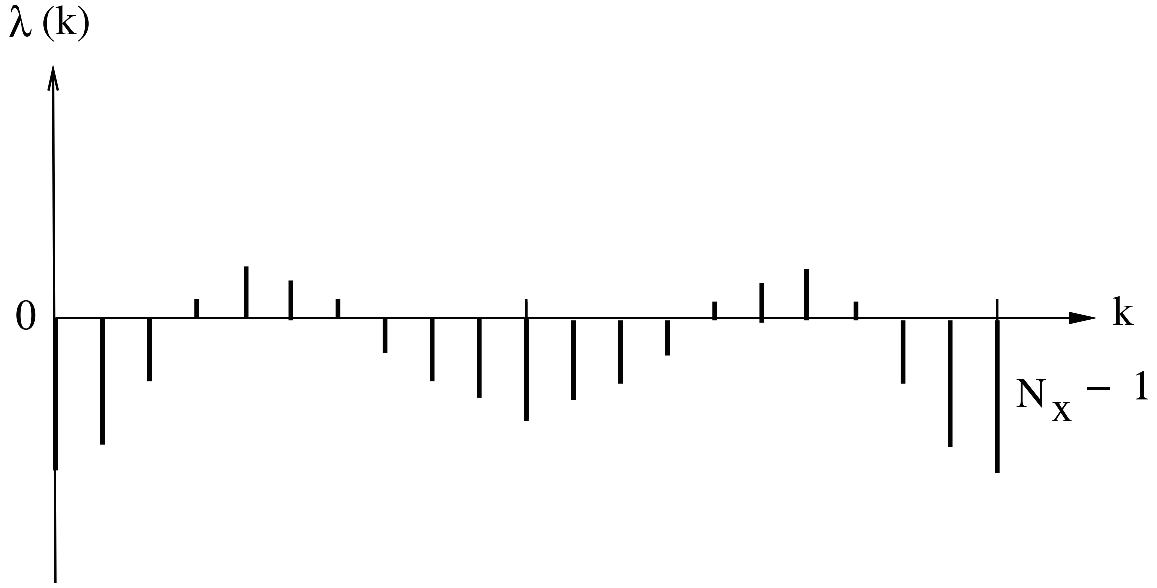


Figure 2.1: Plot of the dependence of the dominant eigenvalue on the spatial frequency index for Turing's two-morphogen reaction-diffusion system. Since $\lambda(0) < 0$, the system is stable for $k_x = 0$ and will thus attenuate the homogeneous perturbations. The system is unstable for a band of spatial wave numbers characterized by $\lambda(k_x) > 0$ and will thus amplify the non-homogeneous perturbations containing these spatial harmonics. The dominant harmonic corresponds to the largest eigenvalue.

k_x^*, corresponding to the critical values of D_A and D_I. These relationships can be determined from the expression for the dominant eigenvalue of (2.13):

$$\begin{aligned}
\lambda(k_x) &= \left(\frac{1}{2}\right)\left(r_{11} + r_{22} - 4D_A\sin^2\left(\frac{\pi k_x}{N_x}\right) - 4D_I\sin^2\left(\frac{\pi k_x}{N_x}\right)\right) \\
&+ \left\{\left[\left(\frac{1}{2}\right)\left(r_{11} + r_{22} - 4D_A\sin^2\left(\frac{\pi k_x}{N_x}\right) - 4D_I\sin^2\left(\frac{\pi k_x}{N_x}\right)\right)\right]^2\right. \\
&- \left[r_{11}r_{22} - r_{12}r_{21} - 4r_{22}D_A\sin^2\left(\frac{\pi k_x}{N_x}\right) - 4r_{11}D_I\sin^2\left(\frac{\pi k_x}{N_x}\right)\right. \\
&+ \left.\left. 16D_AD_I\sin^4\left(\frac{\pi k_x}{N_x}\right)\right]\right\}^{\frac{1}{2}}. \qquad (2.15)
\end{aligned}$$

The first requirement is that the system should be unaffected by homogeneous perturbations. This leads to the following conditions that guarantee $\mathrm{Re}[\lambda(k_x)] < 0$, and thereby ensure the system's stability for all k_x in the case of zero diffusion:

$$r_{11} + r_{22} < 0 \ \text{ and } \ r_{11}r_{22} - r_{12}r_{21} \ > \ 0.$$

The second requirement is that the system should be unstable to non-homogeneous perturbations at least for one value of k_x. An examination of (2.15) reveals that this is possible only if:

$$\begin{aligned} r_{11}r_{22} - r_{12}r_{21} \ &- \ 4r_{22}D_A \sin^2\left(\frac{\pi k_x}{N_x}\right) \\ &- \ 4r_{11}D_I \sin^2\left(\frac{\pi k_x}{N_x}\right) + 16 D_A D_I \sin^4\left(\frac{\pi k_x}{N_x}\right) < 0. \end{aligned} \tag{2.16}$$

The left hand side of (2.16) reaches the minimum when

$$\sin^2\left(\frac{\pi k_x}{N_x}\right) \ = \ \frac{r_{22}D_A + r_{11}D_I}{8D_A D_I}. \tag{2.17}$$

At the edge of instability, (2.16) will be barely satisfied even with this optimal value of $\sin^2\left(\frac{\pi k_x}{N_x}\right)$. Hence, using (2.17) and setting $\mu \stackrel{\text{def}}{=} \frac{D_I}{D_A}$ turns (2.16) into:

$$\mu^2 - 2\left(\frac{r_{11}r_{22} - 2r_{12}r_{21}}{r_{11}^2}\right)\mu + \left(\frac{r_{22}}{r_{11}}\right)^2 \ > \ 0, \tag{2.18}$$

whose solution for the critical value of μ is:

$$\mu \ > \ \frac{r_{11}r_{22} - 2r_{12}r_{21}}{r_{11}^2} + \left[\left(\frac{r_{11}r_{22} - 2r_{12}r_{21}}{r_{11}^2}\right)^2 - \left(\frac{r_{22}}{r_{11}}\right)^2\right]^{\frac{1}{2}}. \tag{2.19}$$

In the mathematical biology community, this equation is known as the condition for the "diffusional", or "Turing", instability. The system is stable if no diffusion is taking place, but if (2.19) is satisfied, then a sufficient flux of morphogens makes the system unstable to non-homogeneous perturbations by (2.16). Intuitively, one often thinks of diffusion as a phenomenon that smoothes temperature and concentration gradients and brings stability. Counter to common intuition, however, reaction-diffusion systems are purposefully set up in such a way that diffusion is necessary in order to cause instability. A possible intuitive explanation for the diffusional instability can be stated as follows. Without the diffusion, there is only enough morphogen to sustain a pilot reaction, and then the system is stable. However, when the morphogen concentration is non-homogeneous, the diffusion "squirts in some extra morphogen", which "fuels" the reaction. Because reaction of a properly formulated reaction-diffusion system is auto-catalytic, the morphogen concentration explodes, thereby driving the system unstable.

Once μ is chosen, (2.17) gives the chemical wavelength, corresponding to the spatial harmonic with the fastest growing amplitude. The chemical wavelength manifests itself as the prominent textural feature on an animal's fur or skin. Solving (2.16) for $\sin^2\left(\frac{\pi k_x}{N_x}\right)$ yields the entire range of spatial frequencies that can cause instability for a given μ:

$$\frac{r_{22}D_A + r_{11}D_I}{8D_AD_I} - \left[\left(\frac{r_{22}D_A + r_{11}D_I}{8D_AD_I}\right)^2 - \frac{r_{11}r_{22} - r_{12}r_{21}}{16D_AD_I}\right]^{\frac{1}{2}} < \sin^2\left(\frac{\pi k_x}{N_x}\right) < \frac{r_{22}D_A + r_{11}D_I}{8D_AD_I} + \left[\left(\frac{r_{22}D_A + r_{11}D_I}{8D_AD_I}\right)^2 - \frac{r_{11}r_{22} - r_{12}r_{21}}{16D_AD_I}\right]^{\frac{1}{2}}. \quad (2.20)$$

2.4.4 Chemical Wavelength Of Hydranth

Now we use (2.14) – (2.20) to analyze the behavior of Turing's actual model for the hydranth regeneration in tubularia (refer to Figure 1.3 . From (2.11), we read the marginal reaction rates:

$$r_{11} = 3,\ r_{12} = 4,\ r_{21} = -4,\ r_{22} = -4. \quad (2.21)$$

These values give $r_{11} + r_{22} = -1$ and $r_{11}r_{22} - r_{12}r_{21} = 4$, thereby satisfying (2.16). If D_A is set to unity, then (2.19) dictates that $D_I > 4$. Substituting $D_I = 4$ into (2.17) gives:

$$\sin^2\left(\frac{\pi k_x}{N_x}\right) = \frac{1}{4}. \tag{2.22}$$

For the given marginal reaction rates and ratio of diffusion constants, the spatial frequency index satisfying this expression is the only one that can cause instability. Therefore, the dominant spatial wave number is:

$$k_x^* = \frac{N_x}{6}. \tag{2.23}$$

Finally, we obtain the characteristic chemical wavelength for the 1-D Turing reaction-diffusion system:

$$\frac{2\pi}{\dfrac{2\pi}{N_x} k_x^*} = 6 \text{ cells}, \tag{2.24}$$

regardless of N_x, the period of the lattice.

As the amplitude of the dominant mode grows, the linear analysis ceases to be valid. However, Turing argues that the linear behavior predicts the overall non-linear behavior reasonably well, because by the time the non-linear effects take over, the supply of morphogens is preempted, and the pattern stops changing. Bard's computer simulations indicate that this is generally true for the 1-D case, but not straightforwardly so for the 2-D case [4], [24].

Note that Turing derived the particular form of the reaction functions in (2.4) along with the values of diffusion constants from biochemical arguments for the case of the hydranth. This means that the regular tentacles, seen in the hydranth, are the manifestations of peaks and valleys in the morphogen concentration, present during the hydranth's development. The spacing of these peaks and valleys, observed in the hydranth, generally agrees with the chemical wavelength of 6 cells, predicted by the theory.

2.5 Pattern-Forming Property Of Reaction-Diffusion And Related Systems

The discussion leading to Table 2.1 emphasizes that only a limited subset of all the possible modes of operation of a reaction-diffusion system are capable of setting up and sustaining spatial waves. This implies that our quest for textures places certain restrictions on the model's parameters.

As our analysis has shown, a reaction-diffusion system can synthesize non-trivial images, provided that it exhibits the Turing instability. And the actual test for the presence of the Turing instability in a reaction-diffusion system is performed on the corresponding linear-reaction reaction-diffusion system, which we obtained by linearizing the original system near its equilibrium. Looking ahead, Chapter 3 introduces the convolutionally-coupled M-lattice system as a generalization of the linear-reaction reaction-diffusion system, and Chapter 4 introduces the new M-lattice system, based on both the convolutionally-coupled M-lattice system and on (2.3), a reaction-diffusion system whose linear interactions are not restricted to the standard diffusion. Since the Turing instability is the instability caused by diffusion, we need a more general definition for evaluating the pattern-forming aspects of these more inclusive models.

Definition 2.5.1 *Linearize the dynamical system near a fixed point and suppose that the resulting small-signal dynamical system is stable to homogeneous perturbations and unstable to non-homogeneous perturbations* [5]*. Then the original system is said to possess the pattern-forming property (see Figure 2.2).*

---

[5] The terms "perturbation" and "evocator" are used interchangeably to denote $\vec{q}(\vec{x})$.

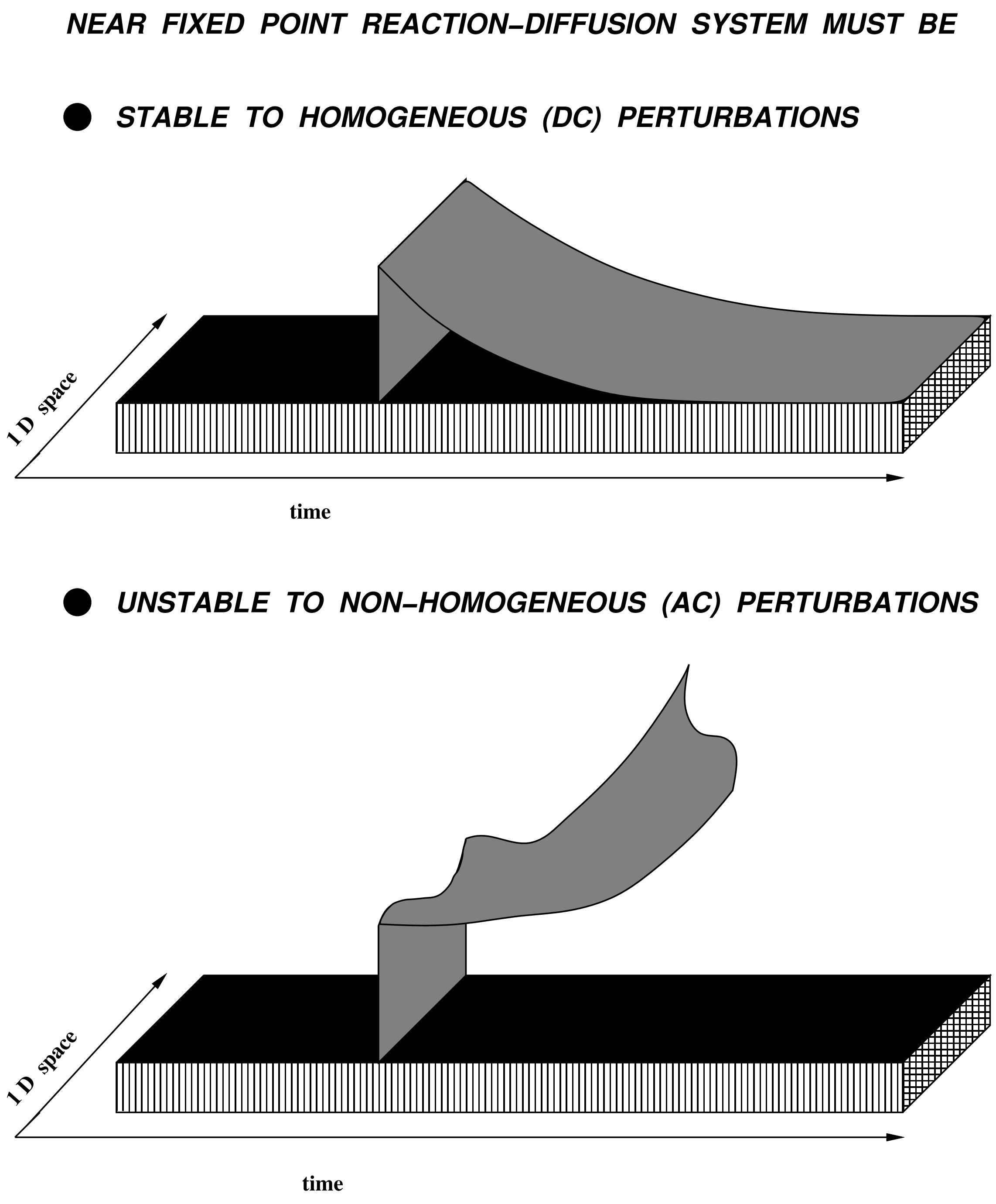


Figure 2.2: An illustration of the pattern-forming property in 1-D. The pattern-forming property is the generalization of the Turing (or diffusional) instability.

It is important to emphasize that stability to homogeneous perturbations is a crucial part of Definition 2.5.1. Without this condition, the constant (DC) level of the morphogen concentration profile can grow without bound, overshadowing the variations. Since concentrations are plotted as pixel intensities, these variations are responsible for the non-trivial image detail. Suppressing the DC component preserves this detail (or texture).

As we shall see in the coming chapters, the classic reaction-diffusion system is by far not the only dynamical system with the pattern-forming property. Other models, such as the convolutionally-coupled M-lattice system and the M-lattice system, can also synthesize textures.

2.6 Computer Simulations Of Turing's 1-D And 2-D Reaction-Diffusion Systems

This section reviews the steps that have been commonly used by other researchers in order to produce reaction-diffusion textures [14], [15].

Section B.3 reviews some basic techniques for simulating PDEs such as (2.4) numerically on a digital computer. Using these techniques, the 1-D Turing reaction-diffusion system, (2.6), discretized as follows:

$$\begin{aligned}
\psi_A(n_x, n_t+1) &= \psi_A(n_x, n_t) + \Delta t\{\psi_A(n_x, n_t)\cdot\psi_I(n_x, n_t) - \psi_A(n_x, n_t) - 12 \\
&+ D_A[\psi_A(n_x+1, n_t) - 2\psi_A(n_x, n_t) + \psi_A(n_x-1, n_t)]\} + q(n_x), \\
\psi_I(n_x, n_t+1) &= \psi_I(n_x, n_t) + \Delta t\{16 - \psi_A(n_x, n_t)\cdot\psi_I(n_x, n_t) \\
&+ D_I[\psi_I(n_x+1, n_t) - 2\psi_I(n_x, n_t) + \psi_I(n_x-1, n_t)]\}, \\
\psi_A(n_x+N_x, n_t) &= \psi_A(n_x, n_t), \\
\psi_I(n_x+N_x, n_t) &= \psi_I(n_x, n_t),
\end{aligned} \tag{2.25}$$

where n_t is the time index, and Δt is the time step, was simulated to convergence. The time step must be made sufficiently small in order for the approximate discrete-time system (2.25) to faithfully track the actual continuous-time system (2.6). In order to reflect physical constraints, the morphogen concentrations must be kept non-negative. If a morphogen is depleted, then its concentration is fixed by the program at zero for the remainder of the simulation (negative concentrations do not make sense) [6]. The evocator was emulated by a pseudo-random number generator. The plot of morphogen concentrations for $N_x = 60$ cells, $D_I = 4$ appears in Figure 2.3. Notice that the peaks in concentrations are separated by approximately 6 cells, predicted by analysis.

The isotropic 2-D Turing reaction-diffusion system (*i.e.*, with identical diffusion rates in the x and the y directions) using $D_I = 6.25$ was simulated for $N_x = N_y = 64$. The diffusion constant of the inhibitor had to be increased in order to comply with the 2-D equivalent of (2.19) (see Section B.3). For displaying purposes, the values of concentrations were linearly scaled to fit between 0 and 255. Figure 2.4 shows the activator concentration, revealing a pattern similar to leopard spots. To illustrate the reaction-diffusion system's high sensitivity to its parameters, D_I is further increased to $D_I = 16$. This causes the spots to turn into "wiggles", as shown in Figure 2.5. Note that these textures correspond to non-equilibrium systems. Depending on the diffusion constant, it takes approximately 2000 iterations at the time step of 0.01 sec.

This thesis emphasizes the use of reaction-diffusion models for synthesis and analysis of textures, regardless of whether or not every detail of the model considered is biologically plausible. The following remarks explain some of the trends found in the reaction-diffusion literature and reshape them in a way that sets the stage for the

[6] This clipping method destroys the consistency of the finite difference method, used for simulating the system. Since reaction-diffusion systems are designed to be unstable, clipping the morphogen concentrations in order to prevent them from overflowing the machine precision is a major issue. See Chapter 3 and Chapter 4 for how this problem is fixed in the M-lattice system.

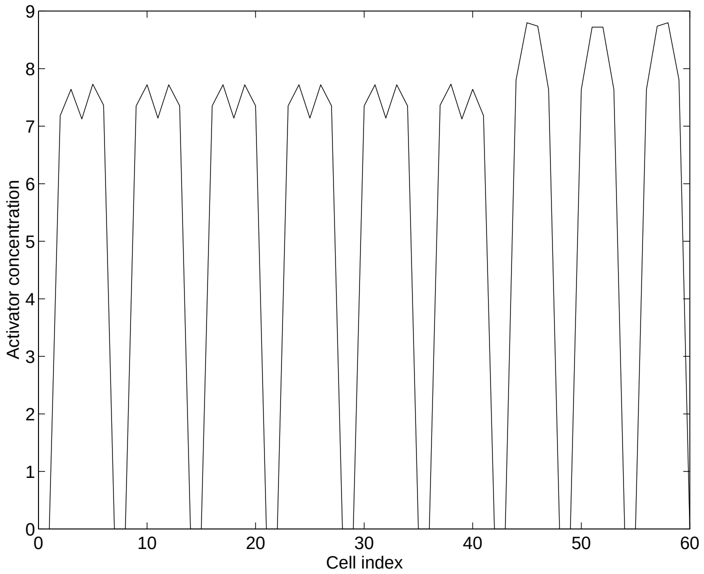


Figure 2.3: Turing's original 's reaction-diffusion model of morphogenesis (for the hydranth). The figure shows the concentration of the activator morphogen as a function of position. The chemical wavelength, predicted by the 1-D linear analysis, can be clearly observed in the plot.

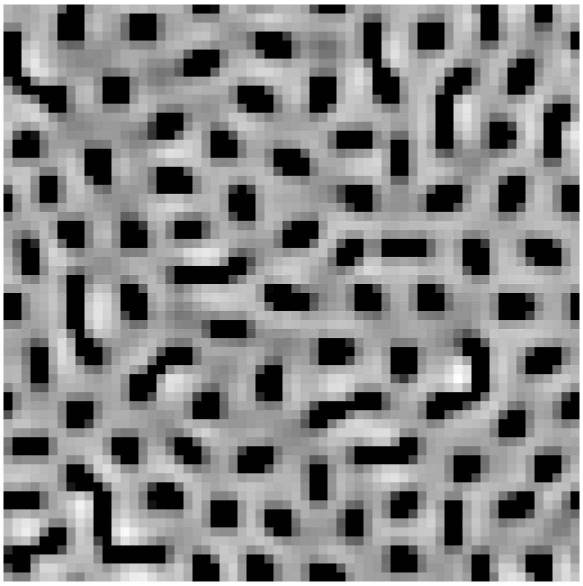


Figure 2.4: Leopard spots modeled by Turing's reaction-diffusion model; $D_I = 6.25$.

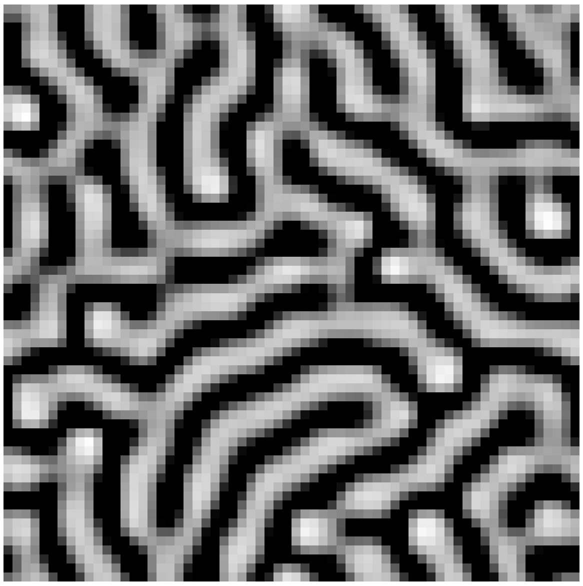


Figure 2.5: "Wiggles" produced by modifying the parameters of Turing's reaction-diffusion model. The diffusion rate of the inhibitor is increased to $D_I = 16$.

remainder of this document.

2.6.1 Parameters For Pattern Formation

This section comments on reaction-diffusion system parameter studies that have been performed by other researchers. A wide range of natural textures have been modeled by RD [4], [24], [8], [9], [6], [10], [7], [13], [15], [14].

Diffusion Rates And Boundary Conditions

Many of these studies involve keeping the same reaction function for all simulations and altering only diffusion rates and boundary conditions in order to produce various textures. We have already seen that diffusion rates have a tremendous effect on the final pattern. In addition, the shape of the 2-D surface on which the simulation is carried out, along with the boundary conditions on that surface, greatly influence the spatial arrangement of the possible kinds of texture. For example, the same reaction-diffusion model can produce long stripes when simulated on the surface of an object shaped like a zebra and short stripes when simulated on a cat's tail [24], [8], [9], [13].

Time Step And Simulation Accuracy

However, some studies erroneously interpreted the time step parameter, Δt, in (2.25) as the speed of reaction [4], [15]. One must keep in mind that the time step is an artificial parameter, used solely for the purposes of simulating continuous-time differential equations numerically on a digital computer. While it is true that a larger Δt speeds up the simulation, this speed has no relation to the speed of chemical reactions in an embryo (*i.e.*, the actual organism – the biological system). Changing the value of Δt affects only the numerical properties of (2.25). For example, the numerical accuracy of solving (2.6) with (2.25) deteriorates as Δt increases. Moreover, if Δt exceeds a certain critical value, then the equations in (2.25) become numerically unstable, producing noisy oscillations of growing amplitude, and thus losing connection with the original equations. Bard studied the speed of pattern formation by generating concentration plots using (2.25) for a wide range of Δt values [4]. However, some of those values lie outside the stability region, making the reported simulations meaningless. As a rule, Δt should be well inside the stability region and small enough for an acceptable accuracy. Assuming that in computer simulation the morphogen concentrations are represented by double-precision floating point numbers, $\Delta t = 0.01$ is adequate for both 1-D and 2-D Turing's reaction-diffusion systems.

2.6.2 A Note On Evocator

We found that the evocator term, $q(\vec{x})$, does not really need to be present as part of the reaction of the activator for the entire time-evolution of (2.4). Simulations show that the behavior of the system is unaltered by simply using $q(\vec{x})$ as the initial condition (*i.e.*, applied only at $n_t = 0$). This minor change puts (2.4) into the convenient standard form of a PDE with initial and boundary conditions:

$$\begin{aligned}
\frac{\partial \psi_m(\vec{x},t)}{\partial t} &= \mathcal{D}_m \psi_m(\vec{x},t) + R_m(\vec{\psi}(\vec{x},t)), \\
R_A(\vec{\psi}(\vec{x},t)) &= \psi_A(\vec{x},t) \cdot \psi_I(\vec{x},t) - \psi_A(\vec{x},t) - 12, \\
R_I(\vec{\psi}(\vec{x},t)) &= 16 - \psi_A(\vec{x},t) \cdot \psi_I(\vec{x},t), \\
\mathcal{D}\psi_A(\vec{x},t) &= D_A \nabla^2 \psi_A(\vec{x},t), \\
\mathcal{D}\psi_I(\vec{x},t) &= D_I \nabla^2 \psi_I(\vec{x},t), \\
\psi_A(\vec{x},t=0) &= \psi_I(\vec{x},t=0) = q(\vec{x}), \\
\psi_A(\vec{x}=\vec{0},t) &= \psi_A(\vec{x}=\vec{N},t), \\
\psi_I(\vec{x}=\vec{0},t) &= \psi_I(\vec{x}=\vec{N},t). \qquad (2.26)
\end{aligned}$$

2.7 Chapter Summary

As a review of the literature, we formulated a general reaction-diffusion system, presented a linear analysis of the 1-D Turing reaction-diffusion system, developed stability / instability conditions, and included computer simulations of the 1-D and the 2-D Turing reaction-diffusion systems as examples. The new information presented in this section includes a critique of the role of the time step in numerical simulations, the additional finding that the evocator needs to be present only as an initial condition, and a working definition of the pattern-forming property for evaluating dynamical systems.

Chapter 3

Linear-Reaction Reaction-Diffusion And M-Lattice Systems

As presented in Chapter 2, reaction-diffusion systems with non-linear reaction are commonly analyzed with the help of linearization. Furthermore, numerical simulations reveal that the linearized equations oftentimes successfully account for the large-signal non-linear behavior. Therefore, a thorough understanding of reaction-diffusion systems with linear reaction is of primary interest. This is the aim of the present chapter, and it contains two main parts:

1. First, we mathematically describe the class of reaction-diffusion systems whose constituent morphogens react linearly.

2. Second, we generalize the concept of a linear-reaction reaction-diffusion system to that of a convolutionally-coupled M-lattice system and solve the latter in closed form. The convolutionally-coupled M-lattice system is a special case of the M-lattice system, the topic of Chapter 4.

3.1 Linear-Reaction Reaction-Diffusion System

3.1.1 Definitions And Notation

Definition 3.1.1 *The general linear-reaction reaction-diffusion system equation is:*

$$\frac{\partial\vec{\psi}(\vec{x},t)}{\partial t} = \mathbf{R}\vec{\psi}(\vec{x},t) + \vec{\mathcal{D}}\vec{\psi}(\vec{x},t), \tag{3.1}$$

where $\mathbf{R} \in \Re^{M\times M}$, $\mathbf{R} = [r_{m_1 m_2}]$ *is the reaction matrix, whose elements,* $r_{m_1 m_2}$*, specify the reaction rates between the morphogens indexed by* m_1 *and* m_2*;* $\mathcal{D}$ *is a general derivative operator of an arbitrary order, just like in (2.3).*

We will develop the convolutionally-coupled M-lattice system by starting with a specific case of $\vec{\mathcal{D}}$, an anisotropic Laplacian, discretized on a 2-D spatial lattice. The anisotropy comes from allowing different weightings of the second derivatives in x and y coordinate axes. The discretization of $\mathcal{D}$ in 2-D is discussed in Section B.3 of Appendix B, and the result for our case is:

$$\begin{aligned}\vec{\mathcal{D}}\vec{\psi}(n_x,n_y,t) &= \mathbf{D}_x[\vec{\psi}(n_x+1,n_y,t) - 2\vec{\psi}(n_x,n_y,t) + \vec{\psi}(n_x-1,n_y,t)] \\ &+ \mathbf{D}_y[\vec{\psi}(n_x,n_y+1,t) - 2\vec{\psi}(n_x,n_y,t) + \vec{\psi}(n_x,n_y-1,t)], \end{aligned} \tag{3.2}$$

where $n_x = 0, \ldots, N_x - 1$ and $n_y = 0, \ldots, N_y - 1$ are the spatial indices, and $\mathbf{D}_x \in \Re^{M\times M}$ and $\mathbf{D}_y \in \Re^{M\times M}$ are the diagonal matrices of diffusion rates in the subscripted dimensions. Therefore, the linear-reaction reaction-diffusion system on a discrete 2-D lattice becomes:

$$\begin{aligned}\frac{\partial\vec{\psi}(n_x,n_y,t)}{\partial t} &= \mathbf{R}\vec{\psi}(n_x,n_y,t) \\ &+ \mathbf{D}_x[\vec{\psi}(n_x+1,n_y,t) - 2\vec{\psi}(n_x,n_y,t) + \vec{\psi}(n_x-1,n_y,t)] \\ &+ \mathbf{D}_y[\vec{\psi}(n_x,n_y+1,t) - 2\vec{\psi}(n_x,n_y,t) + \vec{\psi}(n_x,n_y-1,t)]. \end{aligned} \tag{3.3}$$

3.1.2 Solution

We employ the 2-D DFT for solving (3.3):

$$\begin{aligned}
\vec{\Psi}(k_x, k_y, t) &= \sum_{n_x=0}^{N_x-1} \sum_{n_y=0}^{N_y-1} \vec{\psi}(n_x, n_y, t) \exp\left(-j\frac{2\pi}{N_x} k_x n_x\right) \exp\left(-j\frac{2\pi}{N_y} k_y n_y\right), && (3.4) \\
\vec{\psi}(n_x, n_y, t) &= \frac{1}{N_x N_y} \sum_{k_x=0}^{N_x-1} \sum_{k_y=0}^{N_y-1} \vec{\Psi}(k_x, k_y, t) \exp\left(j\frac{2\pi}{N_x} k_x n_x\right) \exp\left(j\frac{2\pi}{N_y} k_y n_y\right). && (3.5)
\end{aligned}$$

Taking the 2-D DFT, (3.4), of both sides of (3.3) yields:

$$\begin{aligned}
\frac{\partial \vec{\Psi}(k_x, k_y, t)}{\partial t} &= \mathbf{A}(k_x, k_y) \vec{\Psi}(k_x, k_y, t), \\
\mathbf{A}(k_x, k_y) &= \mathbf{R} - 4\left(\sin^2 \frac{\pi k_x}{N_x}\right) \mathbf{D}_x - 4\left(\sin^2 \frac{\pi k_y}{N_y}\right) \mathbf{D}_y, \qquad (3.6) \\
\vec{\Psi}(k_x, k_y, t) &= \exp\left[\mathbf{A}(k_x, k_y) t\right] \vec{\Psi}(k_x, k_y, t = 0).
\end{aligned}$$

The last equation in (3.6) is the solution of the system [1] with the initial condition $\vec{\Psi}(k_x, k_y, t = 0)$.

Formally summarized, the model is (3.6) with the following for $m = 1, \ldots, M$:

$$\det\left[\mathbf{R}\right] \neq 0,\ \forall_m \ \mathrm{Re}\left[\lambda_m(\mathbf{R})\right] < 0 \qquad (3.7)$$

(stability to homogeneous perturbations);

$$\det\left[\mathbf{A}(k_x, k_y)\right] \neq 0,\ \exists_m \ \mathrm{Re}\left[\lambda_m(\mathbf{A}(k_x, k_y))\right] > 0 \qquad (3.8)$$

(instability to non-homogeneous perturbations).

3.2 Convolutionally-Coupled M-Lattice System

The purpose of this section is to show that by introducing some new notation, the linear-reaction reaction-diffusion system can be written as a special case

[1] The reader may recognize this system as a multidimensional form of a first order differential equation.

of a new system, which we call the "convolutionally-coupled M-lattice". The new convolutionally-coupled M-lattice system will be used later in this chapter to analyze some of the existing reaction-diffusion systems and in Chapter 5 to design reaction-diffusion systems, implemented as M-lattice systems, for practical applications. The new notation is based on the DFT, which has been used extensively in the analysis of spatially-discretized PDEs.

3.2.1 Matrix Convolution

We now show how to arrive at the definition of the convolutionally-coupled M-lattice system. The right hand side of the first equation in (3.6) is a multiplication of the matrix, $\mathbf{A}(\vec{k})$, by the vector, $\vec{\Psi}(\vec{k}, t)$, where the notation emphasizes the dependence of the elements of the matrix and the vector on the 2-D spatial frequency, $\vec{k} = [k_x, k_y]^T$. Noting that the diffusion term of (3.2) is a spatial filter on a discrete lattice, we generalize the linear-reaction reaction-diffusion system, (3.3), by replacing the diffusion term with an arbitrary 2-D spatial linear Finite Impulse Response (FIR) filter [25]. Then we study the resulting equations using the Fourier analysis. From linear systems theory, multiplication in the spatial-frequency domain corresponds to convolution in the spatial-index domain [26]. Denoting the 2-D unit sample function by $\delta(\vec{n})$, we propose the following new notation:

$$\mathbf{R}\delta(\vec{n}) \stackrel{\text{def}}{=} \begin{bmatrix} r_{11}\delta(\vec{n}) & \dots & r_{1M}\delta(\vec{n}) \\ \vdots & \vdots & \vdots \\ r_{M1}\delta(\vec{n}) & \dots & r_{MM}\delta(\vec{n}) \end{bmatrix}, \tag{3.9}$$

in order to write:

$$\mathbf{R}\vec{\psi}(n_x, n_y, t) \stackrel{\text{def}}{=} (\mathbf{R}\delta(\vec{n})) * \vec{\psi}(\vec{n}, t). \tag{3.10}$$

Then using (3.10) inside (3.3), we obtain:

$$\begin{aligned}
\frac{\partial\vec{\psi}(\vec{n},t)}{\partial t} &= \mathbf{R}\vec{\psi}(\vec{n},t)+\mathbf{D}(\vec{n})*\vec{\psi}(\vec{n},t) \\
&= (\mathbf{R}\delta(\vec{n}))*\vec{\psi}(\vec{n},t)+\mathbf{D}(\vec{n})*\vec{\psi}(\vec{n},t) \\
&= (\mathbf{R}\delta(\vec{n})+\mathbf{D}(\vec{n}))*\vec{\psi}(\vec{n},t) \\
&= \mathbf{A}(\vec{n})*\vec{\psi}(\vec{n},t), \qquad (3.11)
\end{aligned}$$

where $\mathbf{D}(\vec{n}) \in \Re^{M\times M}$ is a diagonal matrix of the filter coefficients (one convolution mask for each morphogen).

In general, $\mathbf{R}$ can be a full 2-D FIR filter, $\mathbf{R}(\vec{n})$, and $\mathbf{D}(\vec{n})$ does not have to be diagonal. Thus, $\mathbf{A}(\vec{n}) = \mathbf{R}(\vec{n}) + \mathbf{D}(\vec{n})$ becomes the generalization of a single-lattice filter to convolutionally-coupled multiple lattices. The corresponding matrix-vector convolution, defined as follows:

$$\begin{bmatrix} a_{11}(\vec{n}) & \ldots & a_{1M}(\vec{n}) \\ \vdots & \vdots & \vdots \\ a_{M1}(\vec{n}) & \ldots & a_{MM}(\vec{n}) \end{bmatrix} * \begin{bmatrix} \psi_1(\vec{n}) \\ \vdots \\ \psi_M(\vec{n}) \end{bmatrix} = \begin{bmatrix} a_{11}(\vec{n})*\psi_1(\vec{n})+\ldots+a_{1M}(\vec{n})*\psi_M(\vec{n}) \\ \vdots \\ a_{M1}(\vec{n})*\psi_1(\vec{n})+\ldots+a_{MM}(\vec{n})*\psi_M(\vec{n}) \end{bmatrix}, \qquad (3.12)$$

is consistent with the original linear-reaction reaction-diffusion system, (3.11).

Under the DFT, (3.4) a general matrix-vector convolution, (3.12), becomes a general matrix-vector multiplication:

$$\begin{bmatrix} A_{11}(\vec{k}) & \ldots & A_{1M}(\vec{k}) \\ \vdots & \vdots & \vdots \\ A_{M1}(\vec{k}) & \ldots & A_{MM}(\vec{k}) \end{bmatrix} \begin{bmatrix} \Psi_1(\vec{k}) \\ \vdots \\ \Psi_M(\vec{k}) \end{bmatrix} = \begin{bmatrix} A_{11}(\vec{k})\Psi_1(\vec{k})+\ldots+A_{1M}(\vec{k})\Psi_M(\vec{k}) \\ \vdots \\ A_{M1}(\vec{k})\Psi_1(\vec{k})+\ldots+A_{MM}(\vec{k})\Psi_M(\vec{k}) \end{bmatrix}. \qquad (3.13)$$

Applying (3.12) and (3.13) to (3.11) produces:

$$\begin{aligned}
\frac{\partial\vec{\Psi}(\vec{k},t)}{\partial t} &= \mathbf{A}(\vec{k})\vec{\Psi}(\vec{k},t), \\
\mathbf{A}(\vec{k}) &= \mathbf{R}+\mathbf{B}(\vec{k}), \qquad (3.14) \\
\vec{\Psi}(\vec{k},t) &= \exp\left[\mathbf{A}(\vec{k})t\right]\vec{\Psi}(\vec{k},t=0),
\end{aligned}$$

where $\mathbf{B}(\vec{k})$ is the element-by-element DFT of $\mathbf{D}(\vec{n})$.

3.2.2 Developing The Convolutionally-Coupled M-Lattice

Comparing (3.3) with (3.11 and (3.6) with (3.14) makes it clear that the linear-reaction reaction-diffusion system is a special case of a linear coupled-lattice system, in which all interactions are convolutions with linear filters. Generalizing the diffusion to any linear filter allows for a variety of new applications, as we will see in Chapter 5.

Definition 3.2.1 *A coupled-lattice system obeying the following equation is a convolutionally-coupled M-lattice system:*

$$\begin{aligned}\frac{\partial\vec{\psi}(\vec{n},t)}{\partial t} &= \mathbf{A}(\vec{n}) * \vec{\psi}(\vec{n},t),\\ \frac{\partial\vec{\Psi}(\vec{k},t)}{\partial t} &= \mathbf{A}(\vec{k})\vec{\Psi}(\vec{k},t),\\ \vec{\Psi}(\vec{k},t) &= \exp\left[\mathbf{A}(\vec{k})t\right]\vec{\Psi}(\vec{k},t=0),\end{aligned} \tag{3.15}$$

where $\mathbf{A}(\vec{n})$ is of arbitrary form.

The origin of the name "M-lattice" comes from its roots in the reaction-diffusion paradigm, where M stands for the number of morphogens, or layers, in the lattice. This is illustrated in Figure 3.1.

3.2.3 Analyzing Turing's Linearized Model As Convolutionally-Coupled M-Lattice

We now use the theory developed in Section 3.2.1 to gain further insight into the linearized version of Turing's two-morphogen model. This problem is thoroughly treated in Appendix C. Here, we state the main points.

The pattern-forming property, stated in Definition 2.5.1, requires the system in (3.15) to be unstable for one or more non-zero values of $\vec{k}$. This, in turn, implies

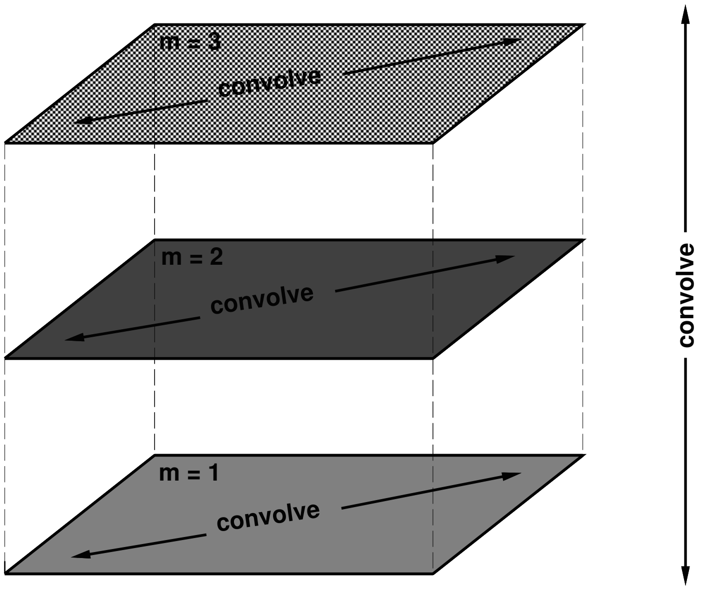


Figure 3.1: Spatial organization of the layers of the convolutionally-coupled M-lattice system. All the intra-layer and the inter-layer interactions are convolutions. Here $M = 3$.

that $\mathbf{A}(\vec{k})$ must have an eigenvalue with a positive real part. For the $M = 2$ case, this is assured by satisfying either or both of the following conditions:

1. The trace of $\mathbf{A}(\vec{k})$ is positive.

2. The determinant of $\mathbf{A}(\vec{k})$ is negative.

Notice that the first condition cannot be satisfied by a system whose $\vec{\mathcal{D}}$ operator is just the standard diffusion (as opposed to a general FIR filter; see Chapter 1). While the standard diffusion can produce only one eigenvalue with a positive real part, the full FIR filter case can produce two. Correspondingly, there are more possibilities for (3.15) to form spatial patterns than for the linear-reaction reaction-diffusion system, (3.1).

Transfer Function Of Convolutionally-Coupled M-Lattice System

Consider (3.15) for a fixed length of time, t. During that time interval, the convolutionally-coupled M-lattice system can be viewed as an M-variable 2-D linear shift-invariant system. Then, in the language of the linear systems theory, $\vec{\Psi}(\vec{k}, t=0)$ is the DFT of the input signal, $\vec{\Psi}(\vec{k}, t)$ is the DFT of the output signal, and the matrix $\exp\left[\mathbf{A}(\vec{k})t\right]$ plays the role of a transfer function.

However, there are fundamental differences between 1-variable and M-variable systems. In 1-variable systems, the transfer function is the DFT of the unit-sample response of the entire system, whereas in M-variable systems, every element of the transfer matrix is a 1-variable transfer function relating one of the inputs to one of the outputs.

Nevertheless, for certain special systems, $\exp\left[\mathbf{A}(\vec{k})t\right]$ contains redundant information, which can be ignored. For example, the input signal to the two-morphogen linear-reaction reaction-diffusion system is restricted in its functional form, because it has the meaning of the evocator (see Section 2.6.2). The evocator signal, $q(\vec{k})$, can be introduced to the system as one of three possible perturbations:

- Only the activator is perturbed: $\vec{\psi}(\vec{k}, t=0) = \left[q(\vec{k})\ 0\right]^T$.
- Only the inhibitor is perturbed: $\vec{\psi}(\vec{k}, t=0) = \left[0\ q(\vec{k})\right]^T$.
- Both morphogens are perturbed by the same signal: $\vec{\psi}(\vec{k}, t=0) = \left[q(\vec{k})\ q(\vec{k})\right]^T$.

For all of these types of the initial condition, the transfer function is a 2-variable vector, but is constructed and used differently.

Appendix C analyzes all three cases. As an example, we state here the equivalent time-dependent transfer function of the two-morphogen linear-reaction reaction-diffusion system for the case when the initial condition is the same for both morphogens.

$$\vec{H}(k_x, k_y, t) \approx \begin{bmatrix} \dfrac{\exp(\lambda_1\, t)\, \left(2\, a_{12} + \sqrt{-4\,\beta + {\tau_e}^2} + \tau_o\right)}{2\,\sqrt{-4\,\beta + {\tau_e}^2}} \\ \dfrac{\exp(\lambda_1\, t)\, \left(2\, a_{21} + \sqrt{-4\,\beta + {\tau_e}^2} - \tau_o\right)}{2\,\sqrt{-4\,\beta + {\tau_e}^2}} \end{bmatrix}, \tag{3.16}$$

where β, τ_e, and τ_o are the determinant, the even trace, and the odd trace, respectively, of $\mathbf{A}(\vec{k})$ as defined in Appendix C. By eliminating the redundant information from $\exp\left[\mathbf{A}(\vec{k})t\right]$, we are able to give $\vec{H}(k_x, k_y, t)$ the usual interpretation of a transfer function. Each element of $\vec{H}(k_x, k_y, t)$ is the DFT of the unit-sample response for the corresponding morphogen.

Due to its exponential growth in time, the transfer function of the linear-reaction reaction-diffusion system is monotonically growing in amplitude for large t (*i.e.*, after the transients have died out). This implies that in the limit as $t \to \infty$, the linear-reaction reaction-diffusion system approaches an ideal band-pass (or "notch") filter. We can think of the transfer function of the linear-reaction reaction-diffusion system as an "AND" operation between the frequency response of the exponentiated eigenvalue and the frequency response of the "pre-filter" that shapes it. Then, as long as the pre-filter is bounded and is either positive for all spatial frequency indices or negative for all spatial frequency indices, the response of the linear-reaction reaction-diffusion system is shaped by the frequency response of the larger eigenvalue. The wave numbers, for which this eigenvalue is positive, will be amplified as a function of time, while the wave numbers, for which this eigenvalue is negative, will be attenuated as a function of time.

For example, consider the case where the transfer function of the linear-reaction reaction-diffusion system is positive for all spatial frequency indices at time zero. A band of wave numbers, for which the eigenvalue is positive, will be amplified continuously with time. Suppose we are designing a notch filter whose purpose is to select a single spatial frequency index. This situation is shown in Figure 3.2. The linear-reaction reaction-diffusion system can be designed to have a positive eigenvalue at that particular wave number and a negative one for all the rest. This means that the linear-reaction reaction-diffusion system will drive the amplitudes of all spatial frequency indices to zero, except for the particular wave numbers of interest, whose amplitude will be increased. For a sufficiently long t, this results in an FIR filter with an extremely large attenuation of the unwanted spatial frequency indices. A traditional FIR filter, designed to accomplish the same feat, will tend to have a large spatial mask – the size of the whole image; however, the linear-reaction reaction-diffusion system can do this with just six parameters (four marginal reaction rates and two diffusion constants). In effect, for a given quality of filtering, the linear-reaction reaction-diffusion system trades off time for the number of filter coefficients. In Chapter 5, we will apply this property to pattern extraction in fingerprint images.

Inverse Visual Appearance Of Morphogens

It has been mentioned in the literature that the plots of the activator and the inhibitor appear as the negatives of one another [4], [24]. This is illustrated in Figure C.5. To our knowledge, the derivation presented in Appendix C is the first published mathematical demonstration of this phenomenon [3].

Separable Versus Non-Separable Filtering

Observe that the diffusion mask that has been used in reaction-diffusion systems has non-zero coefficients only on the x and the y axes. Hence, the DFT of this mask does not have terms that mix the spatial frequencies in the two principal directions. However, because the two morphogens interact within the system, the equivalent reaction-diffusion system filter does have cross terms, albeit in a restricted fashion. We now show that the two-morphogen linear-reaction reaction-diffusion system, whose diffusion filters are separable in 2-D, exhibits a non-separable 2-D filtering behavior.

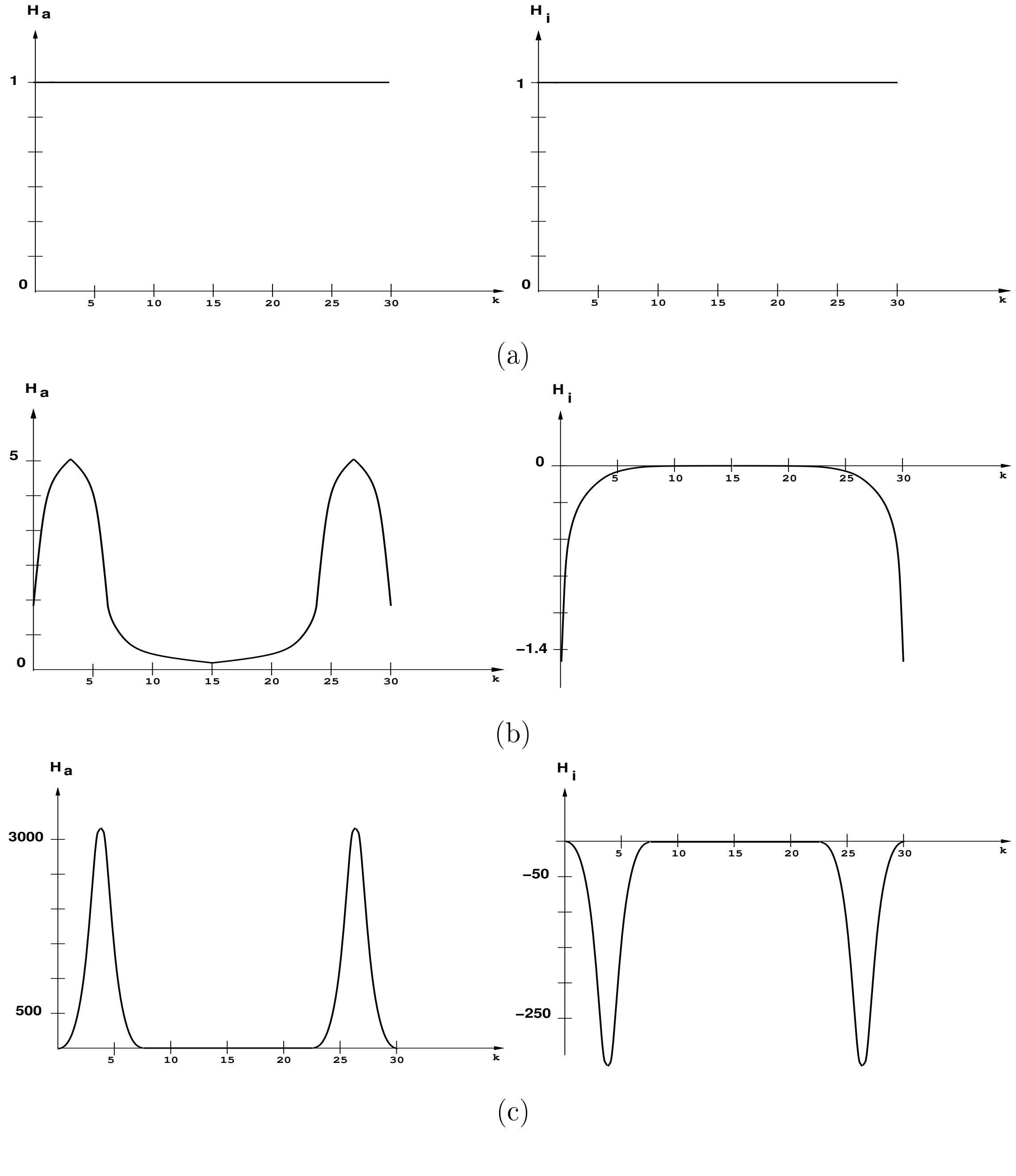


Figure 3.2: Time-dependent transfer function of the linear 1-D Turing's 2-lattice with $N_x = 32$ and a unit sample applied to both morphogens as the evocator input. The temporal snapshots of the transfer function illustrate how the system evolves into a notch filter. The labels H_a and H_i refer to the activator and the inhibitor morphogen components of the transfer function, respectively. (a) the transfer function at $t = 0$ sec; (b) the transfer function at $t = 1$ sec; (c) the transfer function at $t = 5$ sec.

To see this, examine the diffusion terms of $\exp\left[\mathbf{A}(\vec{k})t\right]$ in (C.6):

$$\begin{aligned} B_1(k_x,k_y) &= -4\left(\sin^2\frac{\pi k_x}{N_x}\right)D_{1,x} - 4\left(\sin^2\frac{\pi k_y}{N_y}\right)D_{1,y} \quad \text{and} \qquad (3.17)\\ B_2(k_x,k_y) &= -4\left(\sin^2\frac{\pi k_x}{N_x}\right)D_{2,x} - 4\left(\sin^2\frac{\pi k_y}{N_y}\right)D_{2,y}. \qquad (3.18)\end{aligned}$$

The last term in (C.6) causes the mixing of the wave numbers, k_x and k_y:

$$16\left(D_{1,x}D_{2,y} + D_{1,y}D_{2,x}\right)\left(\sin^2\frac{\pi k_x}{N_x}\right)\left(\sin^2\frac{\pi k_y}{N_y}\right). \qquad (3.19)$$

Distortion In Time-Varying FIR Notch Filter

Since (3.16) has no poles, this linear-reaction reaction-diffusion system is a spatial FIR filter with the marginal reaction rates, the diffusion constants, and the time serving as design parameters. However, the properties of this filter are different from those of traditional FIR filters. A distortion of the initially set up combination of spatial frequencies occurs as a function of time, due to the exponential behavior. Unless the positive eigenvalue stays extremely flat in the desired band of frequencies, the system, with time, turns into the above described notch filter. As time increases, the frequency with the largest eigenvalue will eventually dominate.

3.2.4 The One-Morphogen (Or Reactionless) Case

In addition to the above mentioned differences, there is another important distinction between the general FIR case and the standard diffusion. Since the diffusion coefficients have meaning only when they are non-negative, there is no possibility of achieving a positive eigenvalue without reaction. However, since there is no non-negativity restriction on the FIR filter coefficients, reaction is not needed for a two-morphogen linear-reaction reaction-diffusion system to reach instability [2]. Since

[2]Mentioning FIR filters and instability in the same context might at first appear absurd, since FIR filters are always stable. But the discussion is consistent, since the FIR filters operate in the spatial domain, and the instability of interest takes place in the temporal domain.

we have not seen a formulation of this before, we now show that with an arbitrary FIR filter instead of diffusion, instability to non-homogeneous perturbations can be obtained even with only one morphogen. In other words, a two-morphogen reactionless linear-reaction reaction-diffusion system is equivalent to a one-morphogen linear-reaction reaction-diffusion system.

With all the reaction terms set identically to zero, (C.4) becomes:

$$\begin{aligned}
\lambda_{1,2}(\mathbf{A}(\vec{k})) &= \frac{\mathrm{trace}[\mathbf{B}(\vec{k})]}{2} \pm \left[\left(\frac{\mathrm{trace}[\mathbf{B}(\vec{k})]}{2} \right)^2 - B_1(\vec{k})B_2(\vec{k}) \right]^{\frac{1}{2}} \qquad (3.20) \\
&= \frac{B_1(\vec{k})}{2} + \frac{B_2(\vec{k})}{2} \pm \left[\frac{B_1^2(\vec{k}) + B_2^2(\vec{k})}{2} \right]^{\frac{1}{2}}. \qquad (3.21)
\end{aligned}$$

Either of the two conditions stated in Section 3.2.3 can be easily satisfied by the general FIR filter matrix, $\mathbf{B}(\vec{k})$. Thus, instability is assured if $B_i(\vec{k}) > 0$. The spatial frequencies for which this holds will have growing amplitudes, while the rest will have decaying (or unchanging) amplitudes. The convolutionally-coupled M-lattice system equations become:

$$\begin{aligned}
\frac{\partial \Phi(\vec{k})}{\partial t} &= B(\vec{k})\Phi(\vec{k}), \\
\Phi(\vec{k}) &= \exp\left[B(\vec{k})t\right] \Phi(\vec{k}, t=0).
\end{aligned} \qquad (3.22)$$

In order to determine the conditions under which the one-morphogen system is equivalent to the two-morphogen system we need to compare (3.22) with (C.25), say, for the activator part. Upon equating the respective terms, we obtain:

$$\begin{aligned}
B(\vec{k}) &= \lambda_1(\vec{k}) \;\; \text{and} \qquad (3.23) \\
\Phi(\vec{k}, t=0) &= \left[\frac{2\,a_{12} + \sqrt{-4\,\beta + {\tau_e}^2} + \tau_o}{2\sqrt{-4\,\beta + {\tau_e}^2}} \right] \Psi(\vec{k}, t=0),
\end{aligned}$$

where the quantities on the left refer to the one-morphogen system, and the quantities on the right refer to the two-morphogen system.

Note that in stating these equations, we relied on the fact that the same initial concentrations are given to both the activator and the inhibitor. Hence, the scalar quantity, $\Psi(\vec{k}, t = 0)$, serves as the initial condition.

The transfer function is the quantity that multiplies the input, which is the initial condition. In the case of the full linear-reaction reaction-diffusion system, that quantity is the exponential times the pre-filter, as in (C.25). In the case of the reactionless system, that quantity is just the exponential, as in (3.22). Since the exponential depends on the scalarly weighted time and on no additional terms, it is impossible to make the transfer function of the reactionless system to be identical to that of the full reaction-diffusion system.

3.3 Chapter Summary

In this chapter, we studied the linear-reaction reaction-diffusion system as a first step towards understanding a wider class of reaction-diffusion systems. Toward this goal, we stated the linear-reaction reaction-diffusion system problem and its solution using the $\mathbf{R}$ matrix. In addition, we determined the restrictions on the $\mathbf{R}$ and the $\mathbf{A}(\vec{k})$ matrices to comply with the definition of the pattern-forming property. A central contribution of this chapter is the observation that the linear-reaction reaction-diffusion system is a subset of a more general coupled-lattice linear system, which we call the convolutionally-coupled M-lattice system. We extended the notation of convolution to a matrix-vector case and showed that the convolutionally-coupled M-lattice system can be formulated with the help of this extension. We applied this new technique to the classical two-morphogen linear-reaction reaction-diffusion system and found the exact solutions for the morphogen concentrations as functions of time and spatial indices. We then found the transfer function for the convolutionally-coupled M-lattice system and used it to explain the out-of-phase visual appearance of patterns formed by Turing's reaction-diffusion system. This visual appearance has been reported in the literature, but never explicitly accounted for until the present work. We also used the transfer function to provide quantitative explanations for anisotropy and distortion in filtering associated with the qualitative features of textures synthesized by Turing's reaction-diffusion system. This chapter also states that a single-lattice convolutionally-coupled with arbitrary FIR filters system (which is reactionless) is capable of forming spatial patterns.

Chapter 4

M-Lattice System

This chapter presents the main theoretical contribution of this research – the novel non-linear dynamical system, called "M-lattice". As mentioned in Section 3.2.2, the system's name is inspired by its roots in reaction-diffusion, where M is the number of morphogens, or layers in the lattice. Indeed, the M-layered lattice, or the M-lattice for short, is the last in the chain of models we have been developing in order to analyze and implement the general reaction-diffusion system, (2.3). The linear-reaction reaction-diffusion system, introduced in Chapter 3, is the first model, owing its formulation to the benefits of the linear analysis, presented in Chapter 2. The linear-reaction reaction-diffusion system can be used to analyze the pattern-forming properties of any standard reaction-diffusion system. The second model, the convolutionally-coupled M-lattice system, introduced in Chapter 3 as the generalization of the linear-reaction reaction-diffusion system, facilitates the analysis of pattern-forming properties of any model typified by (2.3).

4.1 Need For Bounded And Continuous Reaction-Diffusion

However, while the convolutionally-coupled M-lattice system is a useful analytical tool, it is impractical. The difficulty is that the pattern-forming dynamical systems, such as reaction-diffusion, are designed to be unstable, and without some form of limiting, the growing sinusoids will eventually overflow machine precision.

It is also desirable for the bounding method to preserve the continuous-time aspect of reaction-diffusion systems. Even though a discrete-time system can faithfully track the time-evolution of its parent continuous-time system, non-linearities generally make discrete-time systems more difficult to analyze mathematically.

4.1.1 Bounding Morphogens In Nature

Note that in morphogenesis, the issue of boundedness does not warrant serious concern because nature takes care of both the lower and the upper bounds automatically. Zero is the lower bound, since negative chemical concentrations do not make sense. The upper bound varies from one biological system to another, but always exists, because an animal's body carries a limited supply of chemicals.

4.1.2 Bounding Morphogens In Engineering Systems

In contrast with nature, where the non-linear limiting of concentrations is caused by the depletion of morphogens, engineering systems require dedicated mechanisms for preventing overflow. We consider three possibilities:

1. Other researchers have forced the limiting (or "clipping") on the morphogen concentrations in computer simulations by adding an "if" statement to the numerical method (*e.g.*, Forward Euler, reviewed in Section B.3.2 of Appendix B) used for solving the system of differential equations [15]. This approach artificially prevents the morphogen concentrations from exceeding the specified bounds. But it also upsets the mathematical consistency of the original dynamical system, because it turns the continuous-time reaction-diffusion model into a discrete-time approximation. Furthermore, the exact functional form of this discrete-time version is different for every numerical method.

2. The primary goal is then to develop a continuous-time system that can control the overflow occurring in reaction-diffusion systems and still be readily analyzed mathematically. The classical approach, based on the the vast body of literature on population dynamics, would encourage an attempt to determine what additional non-linear terms will guarantee boundedness [27], [28]. While many of these diffusionless systems have relied on the non-linear terms to thwart the excessive growth of linearly-unstable solutions [22], the presence of diffusion adds an enormous complexity to the model. This makes the job of finding the non-linearity that will guarantee boundedness and still preserve the pattern-forming property a formidable search. In fact, we are aware of only one success along this avenue of research [29]. Typically, reaction-diffusion systems are derived from biological arguments, with the primary effort aimed at capturing the mechanisms of animal pattern formation, and not at ensuring boundedness [6].

3. We propose a novel method for controlling the morphogen concentrations of reaction-diffusion systems. This method draws inspiration from the way nature handles chemical resources in biological systems (see Section 4.1.1). By using

a "warping" function to facilitate boundedness, the new M-lattice system is continuous in time and allows more flexible non-linear interactions than the reaction-diffusion system [1]. Due to its flexibility, the M-lattice system can be employed to simulate reaction-diffusion as well as many other non-linear dynamical systems. Furthermore, a set of its own mathematical properties, independent of pattern formation, fortifies the M-lattice system with additional modes of operation.

4.2 Mathematical Assumptions

Assume throughout this chapter, unless specified otherwise, that every lemma, proposition, corollary, or any other such statement depends on all the assumptions appearing before that statement. In other words, assumptions accumulate.

4.2.1 Warping Function

Any "sigmoidal" non-linearity can serve as the warping function. The main features of a sigmoidal non-linearity are the low and the high clipping levels and the transition region.

Let $\psi_i(t) \in \Re$ be a state variable as a function of time at each lattice point i, where $i = 1, \ldots, N$. Let $\chi_i(t)$ be an output variable, obtained from $\psi_i(t)$ via $\chi_i(t) = G_i(\psi_i(t))$. The warping functions, $G_i(u)$, can be of different form for different index i. Construct $\vec{\psi}(t)$ and $\vec{\chi}(t)$ by concatenating $\psi_1(t), \ldots, \psi_N(t)$ and $\chi_1(t), \ldots, \chi_N(t)$, respectively, into column vectors.

[1]If the original dynamical system is already bounded on its own, then the warping function performs a "NO-OP" (*i.e.*, it has no effect on the system).

We will use the following assumption in Section 4.3 and and Section 4.4.

Assumption 4.2.1 *Assume that each function, $g_i(u)$, has the following properties:*

- $g_i(u)$ *is continuous on* $u \in \Re$;
- $g_i(u)$ *is real-valued:* $\{g_i : \Re \longrightarrow \Gamma_i,\ \Gamma_i \subset \Re\}$;
- $g_i(u)$ *is odd:* $g_i(u) = -g_i(-u)$;
- $g_i(u)$ *is increasing:* $\dfrac{dg_i(u)}{du} > 0,\ u \in \Re$;
- $g_i(u)$ *is invertible:* $\left\{g_i^{-1} : \Gamma_i \longrightarrow \Re\right\}$; *and*
- $g_i^{-1}(v)$ *is increasing:* $\dfrac{dg_i^{-1}(v)}{dv} > 0,\ v \in \Gamma_i$.

Define each warping function, $G_i(u)$, as a shifted and / or scaled version of $g_i(u)$:

$$G_i(u) = \gamma_i + \alpha_i g_i\left(\frac{u - \gamma_i}{\alpha_i}\right), \tag{4.1}$$

where $\gamma_i,\ \alpha_i \in \Re$ and $\alpha_i \neq 0$.

Remark 4.2.1 *Assumption 4.2.1 implies that* $\inf \Gamma_i = \lim_{u\to-\infty} g_i(u)$ *and* $\sup \Gamma_i = \lim_{u\to+\infty} g_i(u)$*, which means that* $\lim_{v\to\inf\Gamma_i} g_i^{-1}(v) = -\infty$ *and* $\lim_{v\to\sup\Gamma_i} g_i^{-1}(v) = +\infty$.

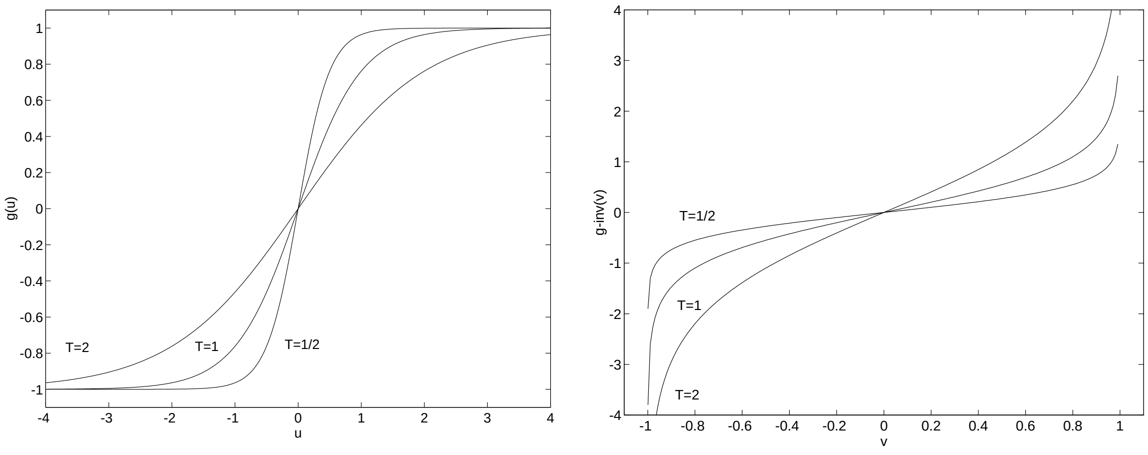
For example, $g_i(u)$ can have the basic form of one of the standard sigmoids, such as the hyperbolic tangent:

$$g(u) = \tanh(\frac{u}{T}), \tag{4.2}$$

$$g^{-1}(v) = T \operatorname{arc}\tanh(v), \tag{4.3}$$

where the value of the "temperature" parameter, $T \in \Re_+$, determines the steepness of the sigmoid. These functions are plotted in Figure 4.1.

In numerical simulation, the warping function is typically applied to each morphogen's concentration after every time step.



(a) (b)

Figure 4.1: Plots of the sigmoidal warping function for three different temperatures. (a) the sigmoid, (4.2); (b) the inverse sigmoid, (4.3).

4.2.2 Assumptions On Comprising Functions

Unless stated otherwise, $\gamma_i = 0 \ \forall i$ and $\alpha_i = 1 \ \forall i$ for simplicity of notation. This does not reduce the generality of (4.1), because all state variables can be shifted and / or scaled if needed.

Assumption 4.2.2 *Assume that the functions, $\left\{a_i(\vec{\psi}(t)) : \Re^N \longrightarrow \Re\right\}$, are defined, continuous, and also differentiable and integrable with respect to $\psi_i(t)$ on any infinite subset of $\Re$.*

Assumption 4.2.3 *Assume that the functions, $\left\{f_i(\vec{\chi}(t)) : \left(\bigcup_{i=1}^N \Gamma_i\right) \longrightarrow \Re\right\}$, are defined, finite, continuous, and also differentiable and integrable with respect to $\chi_i(t)$ on any infinite subset of the open set $\Gamma_i \subset \Re$. Specifically:*

$$\begin{aligned} \exists\ \beta_1 \in \Re_+ : \ \left\|\vec{f}(\vec{\chi}(t))\right\| &\leq \beta_1, \quad \textit{where} \\ \left\|\vec{f}(\vec{\chi}(t))\right\| &= \left(\left[\vec{f}(\vec{\chi}(t))\right]^T \left[\vec{f}(\vec{\chi}(t))\right]\right)^{\frac{1}{2}}. \end{aligned} \tag{4.4}$$

4.2.3 M-Lattice System Definition

Definition 4.2.1 *An M-lattice system is a possibly non-linear autonomous dynamical system, described by the following equation:*

$$\frac{d\psi_i(t)}{dt} = a_i(\vec{\psi}(t)) + f_i(\vec{\chi}(t)), \quad i = 1, \ \ldots, \ N. \tag{4.5}$$

Define $\left\{\vec{a}(\vec{\psi}(t)) : \Re^N \longrightarrow \Re^N\right\}$ *by concatenating* $a_1(\vec{\psi}(t)), \ \ldots, \ a_N(\vec{\psi}(t))$ *into a column vector. Likewise, define* $\left\{\vec{f}(\vec{\chi}(t)) : \left(\bigcup_{i=1}^N \Gamma_i\right) \longrightarrow \Re^N\right\}$ *by concatenating* $f_1(\vec{\chi}(t)), \ \ldots, \ f_N(\vec{\chi}(t))$ *into a column vector. Expressed in a matrix-vector form, (4.5) becomes:*

$$\frac{d\vec{\psi}(t)}{dt} = \vec{a}(\vec{\psi}(t)) + \vec{f}(\vec{\chi}(t)). \tag{4.6}$$

We will use the following assumption in Section 4.3 and Section 4.4.

Assumption 4.2.4 *Assume that one or both of the following are true:*

$$\vec{\nabla}_{\vec{\psi}} \times \frac{d\vec{\psi}(t)}{dt} = \vec{0}; \tag{4.7}$$

$$\vec{\nabla}_{\vec{\chi}} \times \frac{d\vec{\psi}(t)}{dt} = \vec{0}, \tag{4.8}$$

where the subscript indicates the variable, with respect to which the curl is taken [30].

Remark 4.2.2 *Dynamical systems, derived from either formula in Assumption 4.2.4, belong to a general class, called "gradient systems". The name comes from a fundamental theorem in vector calculus, which states that if the curl of a vector field is zero, then that vector field must be the gradient of some "potential" function. Gradient systems are reviewed in Section B.1.4 of Appendix B.*

4.3 General Convergence Proofs

Mathematical background assumed in this section appears in Section B.1 of Appendix B.

Lemma 4.3.1 *The trajectory of the $\psi_i(t)$ (state) and the $\chi_i(t)$ (output) variables of the M-lattice system, (4.5) or (4.6), does not contain cycles.*

Proof: According to Section B.1.4, (4.9) below is a gradient system, and (4.11) below is an almost-gradient system. By Lemma B.1.1 and Corollary B.1.1, respectively, neither system contains cycles. Here, we summarize the essential points of the analysis.

If (4.7) in Assumption 4.2.4 is true, then there exists an auxiliary function, $E(\vec{\psi}(t))$, with the property:

$$\vec{\nabla}_{\vec{\psi}} E(\vec{\psi}(t)) \quad = \quad \frac{d\vec{\psi}(t)}{dt}. \tag{4.9}$$

We compute the time derivative of $E(t) \stackrel{\text{def}}{=} E(\vec{\psi}(t))$ by the chain rule:

$$\frac{dE(t)}{dt} \quad = \quad \left[\vec{\nabla}_{\vec{\psi}} E(\vec{\psi}(t))\right]^T \left[\frac{d\vec{\psi}(t)}{dt}\right] \quad = \quad \left[\frac{d\vec{\psi}(t)}{dt}\right]^T \left[\frac{d\vec{\psi}(t)}{dt}\right] \quad \geq \quad 0. \tag{4.10}$$

Hence, $E(t)$ is strictly increasing, except when $\frac{d\chi_i(t)}{dt} = 0,\ \forall i$ (or $\frac{d\vec{\chi}(t)}{dt} = \vec{0}$), if that ever happens, at which time $E(t)$ reaches a local maximum and stops changing.

If (4.8) in Assumption 4.2.4 is true, then there exists an auxiliary function, $E(\vec{\chi}(t))$, with the property:

$$\vec{\nabla}_{\vec{\chi}} E(\vec{\chi}(t)) \quad = \quad \frac{d\vec{\psi}(t)}{dt}. \tag{4.11}$$

We compute the time derivative of $E(t) \stackrel{\text{def}}{=} E(\vec{\chi}(t))$ by the chain rule:

$$\frac{dE(t)}{dt} \quad = \quad \left[\vec{\nabla}_{\vec{\chi}} E(\vec{\chi}(t))\right]^T \left[\frac{d\vec{\chi}(t)}{dt}\right] \quad = \quad \left[\frac{d\vec{\psi}(t)}{dt}\right]^T \left[\frac{d\vec{\chi}(t)}{dt}\right]$$

$$
\begin{aligned}
&= \sum_i \left\{ \left[\frac{d\psi_i(t)}{dt}\right] \left[\frac{d\chi_i(t)}{dt}\right] \right\} \;=\; \sum_i \left\{ \left[\frac{dG_i^{-1}(\chi_i(t))}{dt}\right] \left[\frac{d\chi_i(t)}{dt}\right] \right\} \\
&= \sum_i \left\{ \left[\frac{dG_i^{-1}(\chi_i(t))}{d\chi_i(t)}\right] \left[\frac{d\chi_i(t)}{dt}\right]^2 \right\} \;\geq\; 0.
\end{aligned} \tag{4.12}
$$

Hence, due to the squared term and Assumption 4.2.1, $E(t)$ is strictly increasing, except when $\frac{d\chi_i(t)}{dt} = 0,\ \forall i$ (or $\frac{d\vec{\chi}(t)}{dt} = \vec{0}$), if that ever happens, at which time $E(t)$ reaches a local maximum and stops changing.

From Assumption 4.2.1 and Assumption 4.2.4, $E(\vec{\psi}(t))$ is a continuous function of the $\psi_i(t)$ (state) variables, and $E(\vec{\chi}(t))$ is a continuous function of the $\chi_i(t)$ (output) variables. This means that, in either case, the curves of constant $E(t)$ (level sets) are closed curves (see Figure 4.2).

The M-lattice system's trajectory (in time) starts at the contour, corresponding to the initial values of the $\psi_i(t)$ (state) and the $\chi_i(t)$ (output) variables. From (4.10) and (4.12), it follows that the trajectory always moves away from the starting contour, without the possibility of return. Moreover, the trajectory moves in the direction of contours with a higher value of $E(t)$, tending toward the nearest local maximum [30]. ∎

Remark 4.3.1 *Lemma 4.3.1 establishes that the trajectories of the M-lattice system's state and output variables do not contain cycles; however, by itself, it provides no information regarding the boundedness of $\vec{\psi}(t)$ or that of either form of $E(t)$.*

Lemma 4.3.2 *The functions, $f_i(\vec{\chi}(t))$, saturate with time. In other words:*

$$
\lim_{t\to\infty} \vec{f}(\vec{\chi}(t)) \;=\; \vec{F}, \tag{4.13}
$$

where $\vec{F}$ is a constant vector.

Proof: The proof follows from Assumption 4.2.1, Assumption 4.2.3, and Lemma 4.3.1. ∎

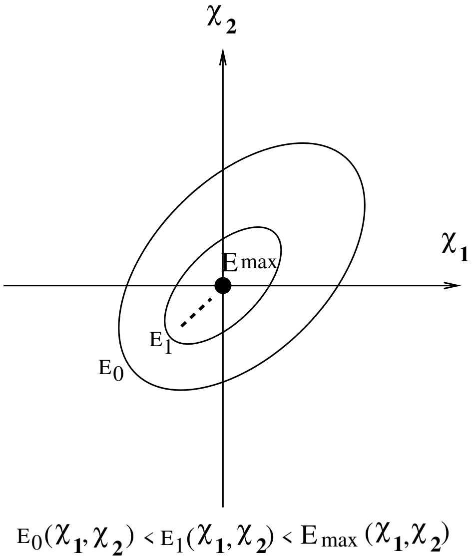


Figure 4.2: The curves of constant $E(\vec{\chi}(t))$. The level sets are closed curves. The trajectory always moves away from the starting contour in the direction of contours with a higher value of $E(t)$.

Assumption 4.3.1 *Assume that all eigenvalues of the Jacobian of $\vec{a}(\vec{\psi}(t))$ are negative at every fixed point of interest, $\vec{\psi}$. Formally:*

$$\forall i \;\; \lambda_i \left[\mathbf{J}_{\vec{a}}(\vec{\psi}) \right] \;\; < \;\; 0, \tag{4.14}$$

where $\mathbf{J}_{\vec{a}}(\vec{\psi}) = \left[\dfrac{\partial a_i(\vec{\psi}(t))}{\partial \psi_j(t)} \right] \Bigg|_{\vec{\psi}(t) = \vec{\psi}}, \quad i,\, j = 1,\, \ldots,\, N.$

Proposition 4.3.1 *Every fixed point of interest, $\vec{\psi}$, of the M-lattice system, (4.5) or (4.6), is locally asymptotically stable.*

Proof: Using Lemma 4.3.2, we rewrite (4.6) in the limit as $t \to \infty$:

$$\lim_{t\to\infty} \frac{d\vec{\psi}(t)}{dt} \;\; = \;\; \lim_{t\to\infty} \vec{a}(\vec{\psi}(t)) + \vec{F}. \tag{4.15}$$

By Assumption 4.2.2, there exists a sufficiently small local neighborhood around every fixed point of interest, $\vec{\psi}$, with the property that in any such local neighborhood, (4.15) becomes:

$$\lim_{t\to\infty} \frac{d\vec{\psi}(t)}{dt} = \lim_{t\to\infty} \mathbf{J}_{\vec{a}}(\vec{\psi})(\vec{\psi}(t) - \vec{\psi}). \tag{4.16}$$

We recognize (4.16) as a linear system for large t. By Assumption 4.3.1, every fixed point of interest of (4.16) is asymptotically stable, implying that (4.16) converges to a constant set of states [31], [23]. Then so does (4.6), provided that the initial condition is sufficiently close to the fixed point of interest. ∎

The following assumption helps strengthen the conclusion of Proposition 4.3.1 and leads to the engineering of practical M-lattice systems.

Assumption 4.3.2 *Assume that $\vec{a}(\vec{\psi}(t))$ in (4.6) is linear in $\psi_i(t)$ (and there is no constant term):*

$$\vec{a}(\vec{\psi}(t)) = \mathbf{A}\vec{\psi}(t), \tag{4.17}$$

where $\mathbf{A} \in \Re^{N\times N}$, $\mathbf{A} = [a_{ij}]$.

Under Assumption 4.3.2, (4.6) turns into the following special case of the M-lattice system:

$$\frac{d\vec{\psi}(t)}{dt} = \mathbf{A}\vec{\psi}(t) + \vec{f}(\vec{\chi}(t)). \tag{4.18}$$

Lemma 4.3.3 *The state vector, $\vec{\psi}(t)$, of the class of M-lattice systems, (4.18) with the previously stated assumptions, is eventually confined to a bounded region. In other words:*

$$\left.\begin{array}{r} \mathcal{B}_r \stackrel{\text{def}}{=} \left\{\vec{\xi} \in \Re^N,\ r \in \Re_+ :\ \left\|\vec{\xi}\right\| < r\right\}, \\ \forall\ \vec{\psi}(t_0)\ \exists\ r = R \in \Re_+ :\ \lim_{t\to\infty} \vec{\psi}(t) \in \overline{\mathcal{B}}_R, \end{array}\right\} \tag{4.19}$$

where $\overline{\mathcal{B}}_R$ is the closure of $\mathcal{B}_R$.

Proof: Let λ_1 be the smallest eigenvalue of $-\mathbf{A}$ and consider the following globally positive-definite function:

$$W(t) \stackrel{\text{def}}{=} W(\vec{\psi}(t)) = \frac{1}{2}\left[\vec{\psi}(t)\right]^T \left[\vec{\psi}(t)\right]. \tag{4.20}$$

From (4.18), the time derivative of $W(t)$ is:

$$\begin{aligned}
\frac{dW(t)}{dt} &= \left[\vec{\psi}(t)\right]^T \left[\frac{d\vec{\psi}(t)}{dt}\right] \\
&= \left[\vec{\psi}(t)\right]^T \left[\mathbf{A}\vec{\psi}(t) + \vec{f}(\vec{\chi}(t))\right] \\
&= \left[\vec{\psi}(t)\right]^T \mathbf{A}\left[\vec{\psi}(t)\right] + \left[\vec{\psi}(t)\right]^T \left[\vec{f}(\vec{\chi}(t))\right] \\
&\le -\lambda_1 \left\|\vec{\psi}(t)\right\|^2 + \beta_1 \left\|\vec{\psi}(t)\right\| \\
&= \left\|\vec{\psi}(t)\right\| \left(-\lambda_1 \left\|\vec{\psi}(t)\right\| + \beta_1\right).
\end{aligned} \tag{4.21}$$

From (4.21), $\dfrac{dW(t)}{dt} < 0$ if $\left\|\vec{\psi}(t)\right\| > \dfrac{\beta_1}{\lambda_1}$. Hence, all solution trajectories eventually enter into the closed ball, $\overline{\mathcal{B}}_R$ (in the $\vec{\psi}$-space) of radius, $R = \dfrac{\beta_1}{\lambda_1}$, centered at $\vec{0}$, and stay there. By Assumption 4.2.1, there is a corresponding compact subset in the $\vec{\chi}$-space, to which all solution trajectories are eventually confined [32], [33], [34], [35]. ∎

Lemma 4.3.4 *If (4.7) in Assumption 4.2.4 is true, then the auxiliary function, $E(t) = E(\vec{\psi}(t))$, defined in (4.9), is a Lyapunov function. Similarly, if (4.8) in Assumption 4.2.4 is true, then the auxiliary function, $E(t) = E(\vec{\chi}(t))$, defined in (4.11), is a Lyapunov function. In other words, either applicable form of $E(t)$ is a Lyapunov function.*

Proof: By Lemma 4.3.1, $E(t)$ is monotonically increasing. In addition, by Lemma 4.3.3, neither the $\psi_i(t)$ (state) variables nor the $\chi_i(t)$ (output) variables escape to infinity. Thus, it remains to show that $E(t)$ is bounded above. Using Lemma 4.3.2, we rewrite (4.18) in the limit as $t \to \infty$:

$$\lim_{t\to\infty} \frac{d\vec{\psi}(t)}{dt} = \lim_{t\to\infty} \mathbf{A}\vec{\psi}(t) + \vec{F}. \tag{4.22}$$

We recognize (4.22) as a driven linear system for large t with the constants, F_i, making up the driving terms. By Assumption 4.3.1 and Assumption 4.3.2, all eigenvalues of $\mathbf{A}$ are negative. Hence, every fixed point of (4.22) is asymptotically stable, so that (4.22) converges to a constant set of states [31], [23]. In other words, $\lim_{t\to\infty} \frac{d\psi_i(t)}{dt} = 0$ (or $\lim_{t\to\infty} \frac{d\vec{\psi}(t)}{dt} = \vec{0}$). This indicates that in the limit as $t \to \infty$, $E(t)$ stops increasing. Therefore, $E(t)$ is a Lyapunov function. ∎

Proposition 4.3.2 *The M-lattice system, (4.5) or (4.6), is totally stable and, therefore, converges in both the $\psi_i(t)$ (state) variables and the $\chi_i(t)$ (output) variables.*

Proof: The proof follows immediately from Lemma 4.3.4. ∎

Corollary 4.3.1 *All fixed points, $\vec{\psi}$, of the M-lattice system, (4.18), are contained in a compact subset of $\Re^N$.*

Proof: Since the set of equilibria of (4.18) consists of isolated points only, it is sufficient to prove that this set is bounded [32]. In other words:

$$\exists \beta_2 \in \Re_+ : \ \left\|\vec{\psi}\right\| \ \leq \ \beta_2. \tag{4.23}$$

Using Assumption 4.3.2 and Proposition 4.3.2, we rewrite (4.18) in the limit as $t \to \infty$:

$$\begin{aligned} \vec{0} &= \lim_{t\to\infty} \mathbf{A}\vec{\psi}(t) + \vec{F}, \quad \text{or} && (4.24)\\ -\lim_{t\to\infty} \mathbf{A}\vec{\psi}(t) &= \vec{F}, \quad \text{or} && (4.25)\\ \lim_{t\to\infty} \vec{\psi}(t) &= -\mathbf{A}^{-1}\vec{F}. && (4.26) \end{aligned}$$

The matrix, $-\mathbf{A}$, is invertible, since it is a positive-definite matrix by Assumption 4.3.1. Then by [35] (4.26) gives:

$$\beta_2 \ = \ \left\|\mathbf{A}^{-1}\right\| \beta_1. \tag{4.27}$$

∎

Remark 4.3.2 *It is important to emphasize the distinction between Proposition 4.3.1 and Proposition 4.3.2. Proposition 4.3.1 states a local property of a fixed point. If the initial condition is sufficiently close to a fixed point of interest, then the solution trajectory approaches it exponentially with time. However, nothing can be said about the solution if the initial condition is not close to the fixed point of interest. The trajectory may head toward another fixed point, get locked into a cycle, or wander off to infinity. On the other hand, Proposition 4.3.2 asserts a property of the whole system. It will converge to some fixed point, regardless of the initial conditions. Various kinds of stability and the significance of total stability are explained in Section B.1 of Appendix B.*

4.4 Examples Of Stable M-Lattice Systems

This section presents two M-lattice systems that satisfy several assumptions of Section 4.3, which ensures their (at least local) asymptotic stability. While we are not aware of any practical uses of the first M-lattice system, we show that the second M-lattice system generalizes the continuous-time continuous-valued (analog) Hopfield network to non-linear objective functions.

Define:

$$\eta_i(t) \stackrel{\text{def}}{=} \int_0^{\psi_i(t)} d\mu G_i(\mu); \tag{4.28}$$

$$\nu_i(t) \stackrel{\text{def}}{=} \int_0^{\chi_i(t)} d\mu G_i^{-1}(\mu). \tag{4.29}$$

For both of the following examples, assume that the given functions, $\Theta(\vec{\psi}(t))$, $\Phi(\vec{\eta}(t))$, $\Theta(\vec{\nu}(t))$, and $\Phi(\vec{\chi}(t))$, are continuous, twice-differentiable, and bounded at least above.

4.4.1 Example 1: Diagonal-Output M-Lattice

Define $E(t)$ as follows:

$$E(t) \stackrel{\text{def}}{=} E(\vec{\psi}(t)) = \Theta(\vec{\psi}(t)) + \Phi(\vec{\eta}(t)). \tag{4.30}$$

Then the M-lattice system becomes:

$$\frac{d\vec{\psi}(t)}{dt} = \vec{\nabla}_{\vec{\psi}} E(\vec{\psi}(t)). \tag{4.31}$$

Clearly, of the two equations in Assumption 4.2.4, (4.31) satisfies (4.7). Explicitly:

$$\frac{d\psi_i(t)}{dt} = \frac{\partial\Theta(\vec{\psi}(t))}{\partial\psi_i(t)} + \frac{\partial\Phi(\vec{\eta}(t))}{\partial\eta_i(t)}\chi_i(t). \tag{4.32}$$

By Proposition 4.3.1, every fixed point of interest of this "diagonal-output" M-lattice system, (4.31) or (4.32), is locally asymptotically stable.

4.4.2 Example 2: Diagonal-State M-Lattice And Hopfield Network

Define $E(t)$ as follows:

$$E(t) \stackrel{\text{def}}{=} E(\vec{\chi}(t)) = \Theta(\vec{\nu}(t)) + \Phi(\vec{\chi}(t)). \tag{4.33}$$

Then the M-lattice system becomes:

$$\frac{d\vec{\psi}(t)}{dt} = \vec{\nabla}_{\vec{\chi}} E(\vec{\chi}(t)). \tag{4.34}$$

Clearly, of the two equations in Assumption 4.2.4, (4.34) satisfies (4.8). Explicitly:

$$\frac{d\psi_i(t)}{dt} = \frac{\partial\Theta(\vec{\nu}(t))}{\partial\nu_i(t)}\psi_i(t) + \frac{\partial\Phi(\vec{\chi}(t))}{\partial\chi_i(t)}. \tag{4.35}$$

By Proposition 4.3.1, every fixed point of interest of this "diagonal-state" M-lattice system, (4.34) or (4.35), is locally asymptotically stable.

Special Case: Hopfield Network

In order for (4.18) to fit the definition of the diagonal-state M-lattice system, (4.34) or (4.35), Assumption 4.3.2 must be simplified as follows:

Assumption 4.4.1 *Assume that* $\mathbf{A}$ *is a diagonal matrix with negative elements on the main diagonal. In other words, let:*

$$\left.\begin{array}{rcl} \dfrac{\partial\Theta(\vec{\nu}(t))}{\partial\nu_i(t)} & = & a_i, \quad \textit{where } \forall i \ a_i < 0; \\ \mathbf{A} & = & \textit{Diag}\ \{a_1,\ \ldots,\ a_N\}, \\ \vec{a}(\vec{\psi}(t)) & = & \mathbf{A}\vec{\psi}(t). \end{array}\right\} \tag{4.36}$$

The resulting diagonal-state M-lattice system becomes:

$$\frac{d\psi_i(t)}{dt} \quad = \quad a_i\psi_i(t) + \frac{\partial\Phi(\vec{\chi}(t))}{\partial\chi_i(t)}. \tag{4.37}$$

By Proposition 4.3.2, this particular diagonal-state M-lattice system is totally stable [35]. Combining (4.29) with (4.37) gives explicit formulae for $\Theta(\vec{\nu}(t))$ and $E(t)$ in (4.33):

$$\Theta(\vec{\nu}(t)) \quad = \quad \sum_i a_i\nu_i(t) = \sum_i \left\{\int_0^{\chi_i(t)} d\mu_i \left[a_i G_i^{-1}(\mu_i)\right]\right\} \quad \text{and} \tag{4.38}$$

$$E(t) \quad = \quad \sum_i a_i\nu_i(t) + \Phi(\vec{\chi}(t)) = \sum_i \left\{\int_0^{\chi_i(t)} d\mu_i \left[a_i G_i^{-1}(\mu_i)\right]\right\} + \Phi(\vec{\chi}(t)). \tag{4.39}$$

The following additional assumption reduces the diagonal-state M-lattice system, (4.37), to the continuous-time continuous-valued Hopfield network [36], [37], [38]:

Assumption 4.4.2 *Assume that* $f_i(\vec{\chi}(t)) \stackrel{\text{def}}{=} \dfrac{\partial\Phi(\vec{\chi}(t))}{\partial\chi_i(t)}$ *in (4.37) is linear in* $\chi_i(t)$*:*

$$\vec{f}(\vec{\chi}(t)) \quad = \quad \vec{\nabla}_{\vec{\chi}}\Phi(\vec{\chi}(t)) \quad = \quad \mathbf{B}\vec{\chi}(t) + \vec{s}, \tag{4.40}$$

where $\mathbf{B} \in \Re^{N\times N}$, $\mathbf{B} = [b_{ij}]$, $\mathbf{B} = \mathbf{B}^T$, *and* $\vec{s} \in \Re^N$.

We summarize the result.

Proposition 4.4.1 *The continuous-time continuous-valued Hopfield network:*

$$\frac{d\psi_i(t)}{dt} = a_i\psi_i(t) + \sum_j b_{ij}\chi_i(t) + s_i \qquad (4.41)$$

is a special case of the diagonal-state M-lattice system. Both systems are totally stable, but the diagonal-state M-lattice system with Assumption 4.4.1 enables the continuous-time continuous-valued Hopfield network to operate with objective functions, $\Phi(\vec{\chi})$, that are not necessarily quadratic in the χ_i (output) variables.

4.4.3 Optimization With Diagonal-State M-Lattice

Since the diagonal-state M-lattice system is totally stable, we have:

$$\begin{aligned}
\psi_i &= \lim_{t\to\infty} \psi_i(t), && (4.42)\\
\vec{\psi} &= \lim_{t\to\infty} \vec{\psi}(t), && (4.43)\\
\chi_i &= \lim_{t\to\infty} \chi_i(t), \quad \text{and} && (4.44)\\
\vec{\chi} &= \lim_{t\to\infty} \vec{\chi}(t). && (4.45)
\end{aligned}$$

Proposition 4.4.2 *Suppose that $\Phi(\vec{\chi})$ is a continuous objective function with continuous partial derivatives with respect to χ_i for $i = 1, \ldots, N$. Let $G_i(u)$ and $g_i(u)$ $\forall i$ be given by (4.1) and (4.2), respectively. Let $T \to 0$ and consider the diagonal-state M-lattice system, (4.34) or (4.35), with Assumption 4.4.1 in the limit as $t \to \infty$. Under these conditions, the diagonal-state M-lattice system seeks to increase $\Phi(\vec{\chi})$ on $\vec{\chi} \in \{-1, 1\}^N$.*

Proof: According to the hypothesis, Proposition 4.3.2 applies. Put $T \to 0$ in (4.2). Then use (4.3) and consider (4.39) in the limit as $T \to 0$ and $t \to \infty$:

$$\begin{aligned}
\lim_{t\to\infty}\lim_{T\to 0} E(t) &= \lim_{t\to\infty}\lim_{T\to 0}\left[\sum_i \left\{\int_0^{\chi_i(t)} d\mu_i \left[a_i G_i^{-1}(\mu_i)\right]\right\} + \Phi(\vec{\chi}(t))\right]\\
&= \lim_{t\to\infty}\left[\lim_{T\to 0}\left(\sum_i \left\{\int_0^{\chi_i(t)} d\mu_i \left[a_i T \operatorname{arc\,tanh}(\mu_i)\right]\right\}\right) + \Phi(\vec{\chi}(t))\right]\\
&= \lim_{t\to\infty} \Phi(\vec{\chi}(t)) \;=\; \Phi(\vec{\chi}). \qquad (4.46)
\end{aligned}$$

By Lemma 4.3.1, $E(t)$ increases. Then (4.46) shows that under the stated conditions the diagonal-state M-lattice system evolves such that $\Phi(\vec{\chi}(t))$ increases as well. ∎

Remark 4.4.1 *It is important to draw distinctions among the functions that are maximized with respect to time, the functions that are maximized with respect to* $\vec{\chi}$*, and the functions that are merely increased. By Lemma 4.3.1, the diagonal-state* M*-lattice system evolves toward a local maximum of* $E(t)$ *with respect to time,* t*. In addition, by (4.12) and Proposition 4.3.2, the system's trajectory tends toward a local maximum of* $E(\vec{\chi})$ *with respect to* $\vec{\chi}$*. When* $T \to 0$*, the system seeks local maxima of* $\Phi(\vec{\chi}(t))$ *with respect to time, since* $E(t) \to \Phi(\vec{\chi}(t))$ *by Proposition 4.4.2. However, it does not seek local maxima of* $\Phi(\vec{\chi})$ *with respect to* $\vec{\chi}$*, since* $\vec{\nabla}_{\vec{\chi}}\Phi(\vec{\chi}) \neq \vec{0}$ *in general. In other words,* $\Phi(\vec{\chi})$ *is merely increased as a function of* $\vec{\chi}$*.*

Nevertheless, there are situations, in which the diagonal-state M*-lattice system can be said to compute local maxima of* $\Phi(\vec{\chi})$ *if the notion of a local maximum is suitably redefined. For example, it has been shown that if* $\dfrac{\partial^2\Phi(\vec{\chi})}{\partial\chi_i^2} = 0\ \forall i$ *and* $T \to 0$*, then the dynamical system seeks local maxima of* $\Phi(\vec{\chi})$ *with respect to* $\vec{\chi}$ *on* $\vec{\chi} \in \{-1, 1\}^N$*, where the locality is defined in terms of the Hamming distance of one [37], [35].*

Remark 4.4.2 *The diagonal-state* M*-lattice system, (4.34) or (4.35), can be compared to the non-linear programming neural network, which optimizes objective functions exclusively on the interior of the* N*-dimensional hypercube [39], [40], [41]. In contrast, the diagonal-state* M*-lattice system seeks to increase the values of objective functions on the corners of the hypercube.*

4.5 Clipped M-Lattice System

In this section, we suspend the continuity, required by Assumption 4.2.1, and consider the M-lattice system, (4.18) with $\vec{f}(\vec{\chi}(t)) \stackrel{\text{def}}{=} \vec{\nabla}_{\vec{\chi}}\Phi(\vec{\chi}(t))$, in which $G_i(u)$ is given by (4.1), but $g_i(u)$ $\forall i$ is the following "clipping" function:

$$\left.\begin{aligned} \frac{dg(u)}{du} &= \begin{cases} \frac{1}{T}, & |u| < T; \\ 0, & |u| \geq T. \end{cases} \\ g(u) &= \frac{1}{2}\left(\left|\frac{u}{T}+1\right| - \left|\frac{u}{T}-1\right|\right); \quad T \in \Re_+. \end{aligned}\right\} \tag{4.47}$$

This function is plotted in Figure 4.3.

For clarity, we provide a formal definition of this new "clipped M-lattice system".

Definition 4.5.1 *Suppose that the given function, $\Phi(\vec{\chi}(t))$, is continuous, twice-differentiable, and bounded at least above. Let $G_i(u)$ and $g_i(u)$ $\forall i$ be given by (4.1) and (4.47), respectively. Then the clipped M-lattice system is the following non-linear dynamical system:*

$$\frac{d\vec{\psi}(t)}{dt} = \mathbf{A}\vec{\psi}(t) + \vec{\nabla}_{\vec{\chi}}\Phi(\vec{\chi}(t)). \tag{4.48}$$

The diagonal-state clipped M-lattice system is the clipped M-lattice system with Assumption 4.4.1.

The topic of the following subsection is the total stability of the diagonal-state clipped M-lattice system, (4.48) with Assumption 4.4.1.

4.5.1 Basic Convergence Proofs

Mathematical background assumed in this section appears in Section B.1 of Appendix B.

Proposition 4.5.1 *The diagonal-state clipped M-lattice system, (4.48) with Assumption 4.4.1, is totally stable and, therefore, converges in both the $\psi_i(t)$ (state) variables and the $\chi_i(t)$ (output) variables.*

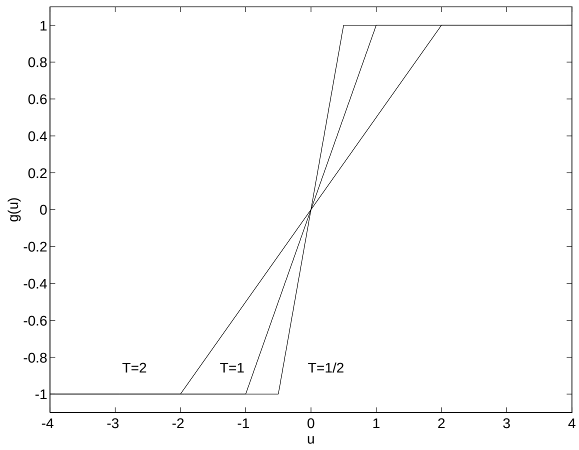


Figure 4.3: Plot of the clipping warping function for three different temperatures.

Proof: The proof will be presented in steps:

1. Find a Lyapunov function, $E(t)$, and prove the stability of every fixed point.

2. Use the continuity of $E(t)$ to prove the asymptotic stability of every fixed point.

Lyapunov Function And Stability

By Lemma 4.3.3, the $\psi_i(t)$ (state) variables and the $\chi_i(t)$ (output) variables of the clipped M-lattice system, (4.48), are bounded. Consider the following auxiliary function for (4.48) with Assumption 4.4.1:

$$E(t) \stackrel{\text{def}}{=} E(\vec{\chi}(t)) = \frac{T}{2}\vec{\chi}(t)^T \mathbf{A}\vec{\chi}(t) + \Phi(\vec{\chi}(t)). \tag{4.49}$$

Define:

$$\mathcal{I} = \{i : |\psi_i| < T \text{ (or } |\chi_i| < 1)\} \tag{4.50}$$

and denote the complement of $\mathcal{I}$ by $\mathcal{I}^c$.

For $E(t)$ to fit the definition of a valid Lyapunov function near a stable fixed point, $\vec{\psi}$ (or $\vec{\chi}$), it is essential that:

$$\vec{\nabla}_{\vec{\psi}} E(\vec{\chi}(t)) = \vec{0} \tag{4.51}$$

at $\vec{\psi}$. From (4.49), $\vec{\nabla}_{\vec{\psi}} E(\vec{\chi}(t))$ is:

$$\begin{aligned}
\vec{\nabla}_{\vec{\psi}} E(\vec{\chi}(t)) &= \mathbf{J}_{\vec{\chi}}(\vec{\psi}(t)) \vec{\nabla}_{\vec{\chi}} E(\vec{\chi}(t)) \\
&= \mathbf{J}_{\vec{\chi}}(\vec{\psi}(t)) \left[T \mathbf{A} \vec{\chi}(t) + \vec{\nabla}_{\vec{\chi}} \Phi(\vec{\chi}(t)) \right] \\
&= \begin{cases} \left(\frac{1}{T}\right) \left(a_i \psi_i(t) + \frac{\partial \Phi(\vec{\chi}(t))}{\partial \chi_i(t)} \right), & i \in \mathcal{I} \\ 0, & i \in \mathcal{I}^c \end{cases} \\
&= \begin{cases} \left(\frac{1}{T}\right) \left(\frac{d\psi_i(t)}{dt} \right), & i \in \mathcal{I} \\ 0, & i \in \mathcal{I}^c. \end{cases}
\end{aligned} \tag{4.52}$$

Since both parts of the last equation in (4.52) identically vanish at a fixed point, the necessary condition in (4.51) is satisfied.

By Assumption 4.3.1, Definition 4.5.1, and (4.47), $E(t)$ is bounded above. In order to show that $E(t)$ is non-decreasing, we observe that according to (4.47), $\mathbf{J}_{\vec{\chi}}(\vec{\psi}(t))$ is a positive-semidefinite matrix. Then differentiating $E(t)$ with respect to time gives:

$$\begin{aligned}
\frac{dE(t)}{dt} &= \left[\vec{\nabla}_{\vec{\psi}} E(\vec{\chi}(t)) \right]^T \left[\frac{d\vec{\psi}(t)}{dt} \right] \\
&= \sum_i \left[\left\{ \begin{array}{ll} \left(\frac{1}{T}\right) \left(\frac{d\psi_i(t)}{dt} \right), & i \in \mathcal{I} \\ 0, & i \in \mathcal{I}^c \end{array} \right\} \left(\frac{d\psi_i(t)}{dt} \right) \right] \\
&= \frac{1}{T} \sum_{i \in \mathcal{I}} \left[\frac{d\psi_i(t)}{dt} \right]^2 \;=\; T \sum_{i \in \mathcal{I}} \left[\frac{d\chi_i(t)}{dt} \right]^2 \;\geq\; 0.
\end{aligned} \tag{4.53}$$

Thus, the auxiliary function, $E(t)$, is bounded above and non-decreasing, and its extrema coincide with the fixed points of the system. By this reasoning, one of the fixed points must correspond to the upper bound or the global maximum of $E(\vec{\psi}(t))$. Other fixed points correspond to the local maxima, the saddles, and the local minima of $E(\vec{\psi}(t))$. The saddles and the local minima of $E(\vec{\psi}(t))$ correspond to unstable fixed points, because $E(t)$ is non-decreasing along the non-trivial solution trajectories. For the same reason, the local maxima of $E(\vec{\psi}(t))$ correspond to stable fixed points, and since $E(\vec{\psi}(t))$ has a global maximum, the system has at least one stable fixed point. Hence, every trajectory of the diagonal-state clipped M-lattice system, (4.48), contains a stable fixed point, $\vec{\psi}$ (or $\vec{\chi}$), and $E(t)$ is a Lyapunov function.

Strengthening Stability To Asymptotic Stability

We now proceed to show that every such $\vec{\psi}$ (or $\vec{\chi}$) is asymptotically stable as well. It is evident from (4.53) that:

$$\lim_{t\to\infty} \frac{dE(t)}{dt} = 0 \iff \lim_{t\to\infty} \frac{d\psi_i(t)}{dt} = \lim_{t\to\infty} \frac{d\chi_i(t)}{dt} = 0, \ \forall i \in \mathcal{I}. \tag{4.54}$$

Observe that since:

$$\frac{dg(\psi_i(t))}{d\psi_i(t)} \quad = \quad 0, \ \forall i \in \mathcal{I}^c, \tag{4.55}$$

$\lim_{t\to\infty} \frac{dE(t)}{dt}$ is unaffected by $\lim_{t\to\infty} \frac{d\psi_i(t)}{dt}$, $i \in \mathcal{I}^c$ (or $\lim_{t\to\infty} \frac{d\chi_i(t)}{dt}$, $i \in \mathcal{I}^c$). In other words, once all $\psi_i(t)$, $i \in \mathcal{I}$ stop changing, $E(t)$ reaches a constant value and stays there forever, regardless of any possible dynamics of $\chi_i(t) \in \{-1, 1\}$, $i \in \mathcal{I}^c$.

The next step is to show that, in fact, no dynamics of $\chi_i(t) \in \{-1, 1\}$, $i \in \mathcal{I}^c$ are possible in the limit as $t \to \infty$. Formally, we must prove that:

$$\lim_{t\to\infty} \chi_i(t) = \chi_i, \ i \in \mathcal{I}^c \quad \text{and hence} \quad \lim_{t\to\infty} \vec{\chi}(t) = \vec{\chi} \iff \lim_{t\to\infty} \frac{d\vec{\chi}(t)}{dt} = \vec{0}. \tag{4.56}$$

Consider the following alternative expression for $\frac{dE(t)}{dt}$ in the limit as $t \to \infty$:

$$\lim_{t\to\infty} \frac{dE(t)}{dt} \quad = \quad \lim_{t\to\infty} \left\{ \left[\vec{\nabla}_{\vec{\chi}} E(\vec{\chi}(t)) \right]^T \left[\frac{d\vec{\chi}(t)}{dt} \right] \right\} \quad = \quad 0, \tag{4.57}$$

where the last equality is due to (4.53) and (4.54). According to (4.57), $\vec{\nabla}_{\vec{\chi}} E(\vec{\chi}(t))$ tends to become orthogonal to the system's trajectory, $\vec{\chi}(t)$, in the limit as $t \to \infty$. This is consistent with (4.53) and means that the trajectory tends toward a level set of $E(t)$. Now, assume that the converse of the second statement in (4.56) is true. Then in the limit as $t \rightarrow \infty$, the trajectory traverses the level set. Thus, taking (4.54) into account, we have:

$$\exists i \in \mathcal{I}^c,\ \exists \tau \in \Re,\ \tau \rightarrow \infty : \ \left.\frac{d\chi_i(t)}{dt}\right|_{t\,=\,\tau} \ \neq \ 0. \tag{4.58}$$

The statement in (4.58) is equivalent to:

$$\exists i \in \mathcal{I}^c,\ \exists \tau \in \Re_+,\ \tau \rightarrow \infty,\ \exists \varepsilon \in \Re_+,\ \varepsilon \rightarrow 0 : \ \chi_i(\tau+\varepsilon) \ = \ -\chi_i(\tau). \tag{4.59}$$

In order to satisfy (4.59), the trajectory, $\vec{\chi}(t)$ or $\vec{\psi}(t)$, of the diagonal-state clipped M-lattice system along the sought level set of $E(t)$ must encounter a discontinuity. But this would violate the continuity of $E(t)$, thereby contradicting Definition 4.5.1. In other words, $\chi_i(t)$ cannot toggle without passing through the continuum of intermediate values, which is impossible in the limit as $t \rightarrow \infty$, for $E(\vec{\psi}(t))$ reaches a local maximum, and $\mathcal{I}$ and $\mathcal{I}^c$ become constant sets (*i.e.*, the counts of indices comprising the sets and the indices themselves stop changing with time). Hence, the conjecture in (4.56) follows.

Effectively, we have shown that the trajectory of the $\chi_i(t)$ (output) variables does not contain cycles, meaning that the conclusions of Lemma 4.3.1 extend to the clipped M-lattice system. Then (4.54) becomes:

$$\lim_{t\rightarrow\infty} \frac{dE(t)}{dt} = 0 \iff \lim_{t\rightarrow\infty} \frac{d\psi_i(t)}{dt} = 0,\ \forall i \in \mathcal{I};\ \lim_{t\rightarrow\infty} \frac{d\vec{\chi}(t)}{dt} = \vec{0}. \tag{4.60}$$

Finally, we invoke (4.47), Definition 4.5.1, and (4.60) in order to adopt Lemma 4.3.2 and Lemma 4.3.4 to the diagonal-state clipped M-lattice system. As a result, (4.54) and (4.60) are strengthened to:

$$\lim_{t\rightarrow\infty} \frac{dE(t)}{dt} = 0 \iff \lim_{t\rightarrow\infty} \frac{d\vec{\psi}(t)}{dt} = \lim_{t\rightarrow\infty} \frac{d\vec{\chi}(t)}{dt} = \vec{0}. \tag{4.61}$$

This completes the proof. ∎

Remark 4.5.1 *If $T = 1$, $\vec{f}(\vec{\chi}(t)) \stackrel{\text{def}}{=} \vec{\nabla}_{\vec{\chi}}\Phi(\vec{\chi}(t))$ is linear in $\chi_i(t)$, $\mathbf{J}_{\vec{f}}(\vec{\psi})$ is a circulant (or block-circulant) and symmetric matrix, and $a_i = -|a|$, $\forall\ i$, then the diagonal-state clipped M-lattice system reduces to the reciprocal cellular neural network [42], [43], [44], [45], [46] [2]:*

$$\frac{d\psi(\vec{n},t)}{dt} = -|a|\,\psi(\vec{n},t) + b(\vec{n}) * \chi(\vec{n},t) + s(\vec{n}) \quad \textit{or} \tag{4.62}$$

$$\frac{d\psi_i(t)}{dt} = -|a|\,\psi_i(t) + \sum_j b_{i,j}\chi_j(t) + s_i. \tag{4.63}$$

Since the clipped M-lattice system, (4.48), permits arbitrary linear interactions among the state variables, non-linear interactions among the output variables, and allows any $T \in \Re_+$, it is a generalization of the diagonal-state clipped M-lattice system and of the reciprocal cellular neural network. However, the clipped M-lattice system is not totally stable for all parameter values.

4.5.2 Designing Clipped M-Lattice System To Have Binary Outputs

Many applications that rely on solving the so-called "coordinated decisions problems" require binary outputs. Digital image halftoning, edge and corner detection, pattern extraction, and texture classification are some examples of coordinated decisions problems in the field of image processing. In this subsection, we derive conditions for driving the outputs of two subclasses of the clipped M-lattice system to the corners of the hypercube. These results are applied in Section 5.3.

[2]The authors of the original cellular neural network paper [42] presented a sketch of the proof of total stability, but it contains errors and omissions.

General Condition For Binary Outputs

For notational convenience, denote $\dfrac{d\vec{\psi}(t)}{dt}$ in (4.48) by $\vec{c}(\vec{\psi}(t))$. Then the following lemma contains the goal of a design procedure that guarantees binary outputs.

Lemma 4.5.1 *Suppose that the clipped M-lattice system, (4.48), is totally stable and consider its evolution towards an asymptotically stable fixed point, $\vec{\psi}$. If $\mathbf{J}_{\vec{c}}(\vec{\psi})$ (the Jacobian of $\vec{c}(\vec{\psi}(t))$ at $\vec{\psi}$) is such that $\exists i$ $Re\left(\lambda_i\left[\mathbf{J}_{\vec{c}}(\vec{\psi})\right]\right) > 0$, then $\mathcal{I} = \emptyset$. In other words, $|\mathcal{I}| = 0$, $|\mathcal{I}^c| = N$, and the stable fixed point is $\vec{\chi} \in \{-1, 1\}^N$.*

Proof: By Assumption 4.2.2, Assumption 4.2.3, and Definition 4.5.1, there exists a sufficiently small local neighborhood around the fixed point, for which (4.48) becomes [30], [47], [48]:

$$\begin{aligned} \frac{d\vec{\psi}(t)}{dt} &= \vec{c}(\vec{\psi}(t)) = \mathbf{J}_{\vec{c}}(\vec{\psi})(\vec{\psi}(t) - \vec{\psi}) \\ &= \left(\mathbf{A} + \mathbf{H}_{\Phi}(\vec{\psi})\mathbf{J}_{\vec{\chi}}(\vec{\psi})\right)(\vec{\psi}(t) - \vec{\psi}), \end{aligned} \tag{4.64}$$

where $\mathbf{H}_{\Phi}(\vec{\psi})$ is the Hessian of $\Phi(\vec{\chi}(t))$ at the fixed point, $\vec{\psi}(t) = \vec{\psi}$:

$$\mathbf{H}_{\Phi}(\vec{\psi}) \in \Re^{N \times N},\ \mathbf{H}_{\Phi}(\vec{\psi}) = \left[h_{ij}(\vec{\psi})\right],\ h_{ij}(\vec{\psi}) = \left.\frac{\partial^2 \Phi(\vec{\chi}(t))}{\partial \chi_i(t) \partial \chi_j(t)}\right|_{\vec{\psi}(t) = \vec{\psi}}. \tag{4.65}$$

Assume that the premise of the total stability holds, but $|\mathcal{I}| > 0$ (*i.e.*, $\mathcal{I} \neq \emptyset$). According to the hypothesis on the eigenvalues of the Jacobian, the linearized (small-signal) system, (4.64), is unstable [31], [23]. Because of this, the overall (large-signal) clipped M-lattice system is also unstable [47], [48]. But this contradicts the assumption of total stability. Hence, the conclusion follows. ∎

Remark 4.5.2 *If $|\mathcal{I}| = 0$ for the general clipped M-lattice system, (4.48), then $\mathbf{H}_{\Phi}(\vec{\psi})\mathbf{J}_{\vec{\chi}}(\vec{\psi}) = \mathbf{0}$, implying that every fixed point of the form, $\vec{\chi} \in \{-1, 1\}^N$, is locally asymptotically stable by Proposition 4.3.1.*

Solution For Circulant Symmetric Matrices

Now, consider a class of the clipped M-lattice system, in which $\mathbf{A}$ and $\mathbf{H}_{\Phi}(\vec{\psi})$ are restricted to be circulant. Such matrices correspond to 1-D finite impulse response (FIR) filters. For 2-D filters, $\mathbf{A}$ and $\mathbf{H}_{\Phi}(\vec{\psi})$ become block-circulant. In a circulant or a block-circulant matrix, the same number appears on the main diagonal. A 1-D filter, $h(n)$, is symmetric if $h(n) = h(N-n)$. For 2-D FIR filters, the symmetry requirement is: $h(n_x, n_y) = h(N - n_x,\ N - n_y)$. If the filter is symmetric, then the corresponding circulant or block-circulant matrix is also symmetric.

For example, circularly convolving with the following symmetric 1-D FIR filter:

$$h(n) = \left\{ \begin{array}{ll} h_0, & n = 0; \\ h_1, & n = 1; \\ h_1, & n = 2 \end{array} \right\} \tag{4.66}$$

is equivalent to multiplying on the left by the following symmetric circulant matrix:

$$H = \begin{bmatrix} h_0 & h_1 & h_1 \\ h_1 & h_0 & h_1 \\ h_1 & h_1 & h_0 \end{bmatrix}. \tag{4.67}$$

Analogously, if the symmetric 2-D FIR filter is:

$$h(n_x, n_y) = \left\{ \begin{array}{ll} h_{00}, & (n_x, n_y) = (0,0); \\ h_{01}, & (n_x, n_y) = (0,1); \\ h_{01}, & (n_x, n_y) = (0,2); \\ h_{10}, & (n_x, n_y) = (1,0); \\ h_{11}, & (n_x, n_y) = (1,1); \\ h_{12}, & (n_x, n_y) = (1,2); \\ h_{10}, & (n_x, n_y) = (2,0); \\ h_{12}, & (n_x, n_y) = (2,1); \\ h_{11}, & (n_x, n_y) = (2,2) \end{array} \right\}, \tag{4.68}$$

then the corresponding symmetric block-circulant matrix is:

$$H = \left[\begin{array}{ccc|ccc|ccc} h_{00} & h_{01} & h_{01} & h_{10} & h_{11} & h_{12} & h_{10} & h_{12} & h_{11} \\ h_{01} & h_{00} & h_{01} & h_{12} & h_{10} & h_{11} & h_{11} & h_{10} & h_{12} \\ h_{01} & h_{01} & h_{00} & h_{11} & h_{12} & h_{10} & h_{12} & h_{11} & h_{10} \\ \hline h_{10} & h_{12} & h_{11} & h_{00} & h_{01} & h_{01} & h_{10} & h_{11} & h_{12} \\ h_{11} & h_{10} & h_{12} & h_{01} & h_{00} & h_{01} & h_{12} & h_{10} & h_{11} \\ h_{12} & h_{11} & h_{10} & h_{01} & h_{01} & h_{00} & h_{11} & h_{12} & h_{10} \\ \hline h_{10} & h_{11} & h_{12} & h_{10} & h_{12} & h_{11} & h_{00} & h_{01} & h_{01} \\ h_{12} & h_{10} & h_{11} & h_{11} & h_{10} & h_{12} & h_{01} & h_{00} & h_{01} \\ h_{11} & h_{12} & h_{10} & h_{12} & h_{11} & h_{10} & h_{01} & h_{01} & h_{00} \end{array}\right]. \tag{4.69}$$

Using symmetric FIR filters as parameters of clipped M-lattice systems is convenient. First, the FIR computation is local in nature. Locality is beneficial, because it minimizes the inter-processor wiring requirement in VLSI realization and the amount of floating point operations in computer simulation. Second, the added symmetry makes the computation not only local, but also modular. Symmetric FIR hardware is a scalable structure, built out of identical computation / communication modules.

Moreover, as we will show, properly choosing just one parameter, $h(\vec{n} = \vec{0})$, is sufficient in order to drive the outputs of the diagonal-state clipped M-lattice system to ± 1. For notational simplicity, we have dropped the dependence of $\mathbf{H}_\Phi(\vec{\psi})$ on $\vec{\psi}$, but it should be clear that the requirements to be derived must hold for any value of $\vec{\psi}$. Simplifying the notation of $\mathbf{H}_\Phi(\vec{\psi})$ to $\mathbf{H}$ is strictly legitimate only when $\mathbf{H}_\Phi(\vec{\psi})$ is a constant matrix.

Thus, the goal is to satisfy the condition of Lemma 4.5.1 by choosing $h(\vec{0})$ such that $\mathbf{J}_{\vec{c}}(\vec{\psi})$ has at least one eigenvalue with a positive real part for $|\mathcal{I}| > 0$.

Proposition 4.5.2 *Assume that the clipped M-lattice system, (4.48), is totally stable. Suppose that* **A** *and* **H** *used in the clipped M-lattice system, (4.48), correspond to symmetric filters, and the following conditions are satisfied:*

$$\begin{cases} a(\vec{0}) + \dfrac{h(\vec{0})}{T} - \sum\limits_{\vec{n}\neq\vec{0}} |a(\vec{n})| > a(\vec{0}) + \left| a(\vec{k}) + \dfrac{h(\vec{k})}{T} \right| + \sum\limits_{\vec{n}\neq\vec{0},\vec{n}\neq\vec{k}} |a(\vec{n})|, \\ a(\vec{0}) + \dfrac{h(\vec{0})}{T} - \sum\limits_{\vec{n}\neq\vec{0}} |a(\vec{n})| > 0, \\ \text{where } \vec{k} = \arg\max\limits_{\vec{n}\neq\vec{0}} \left| a(\vec{n}) + \dfrac{h(\vec{n})}{T} \right|. \end{cases} \tag{4.70}$$

Then $\vec{\chi} \in \{-1, 1\}^N$.

Proof: Introduce additional notation:

$$\mathbf{B} = \mathbf{J}_{\vec{c}}(\vec{\psi}) = \left(\mathbf{A} + \mathbf{H}\mathbf{J}_{\vec{\chi}}(\vec{\psi})\right). \tag{4.71}$$

For the purposes of illustration, we will use the 3×3 example of (4.66) and (4.67):

$$\mathbf{B} = \begin{bmatrix} a_0 + h_0 \dfrac{dg(\psi_1)}{d\psi_1} & a_1 + h_1 \dfrac{dg(\psi_2)}{d\psi_2} & a_1 + h_1 \dfrac{dg(\psi_3)}{d\psi_3} \\ a_1 + h_1 \dfrac{dg(\psi_1)}{d\psi_1} & a_0 + h_0 \dfrac{dg(\psi_2)}{d\psi_2} & a_1 + h_1 \dfrac{dg(\psi_3)}{d\psi_3} \\ a_1 + h_1 \dfrac{dg(\psi_1)}{d\psi_1} & a_1 + h_1 \dfrac{dg(\psi_2)}{d\psi_2} & a_0 + h_0 \dfrac{dg(\psi_3)}{d\psi_3} \end{bmatrix}. \tag{4.72}$$

There is no loss of generality, and the proof is valid for **B** of any size.

Observe that the worst-case situation occurs when only one output variable has not yet saturated to ± 1 (*i.e.*, the trajectory of the clipped M-lattice system is moving along an edge of the N-dimensional hypercube):

$$\mathbf{B}_{\text{edge}} = \begin{bmatrix} a_0 + \dfrac{h_0}{T} & a_1 & a_1 \\ a_1 + \dfrac{h_1}{T} & a_0 & a_1 \\ a_1 + \dfrac{h_1}{T} & a_1 & a_0 \end{bmatrix}. \tag{4.73}$$

The chosen value of h_0 must guarantee that the trajectory will not terminate on the edge, but head to a corner.

In order to see this, consider (4.72) at an interior point:

$$\mathbf{B}_{\text{int}} = \begin{bmatrix} a_0 + \frac{h_0}{T} & a_1 + \frac{h_1}{T} & a_1 + \frac{h_1}{T} \\ a_1 + \frac{h_1}{T} & a_0 + \frac{h_0}{T} & a_1 + \frac{h_1}{T} \\ a_1 + \frac{h_1}{T} & a_1 + \frac{h_1}{T} & a_0 + \frac{h_0}{T} \end{bmatrix}. \tag{4.74}$$

Since $\mathbf{B}_{\text{int}}$ is circulant, it has the same value on the main diagonal. Recall that $\mathbf{A}$ is a circulant and negative-definite matrix, which implies that $a_0 < 0$. Then setting $h_0 > -Ta_0$ gives trace $[\mathbf{B}_{\text{int}}] > 0$. Since $\mathbf{B}_{\text{int}}$ is symmetric with positive trace, it possesses at least one positive eigenvalue. This makes interior fixed points unstable, and some output variable must saturate to ± 1.

With one output variable saturated, the candidate fixed points are on the faces of the hypercube, and:

$$\mathbf{B}_{\text{face}} = \begin{bmatrix} a_0 + \frac{h_0}{T} & a_1 + \frac{h_1}{T} & a_1 \\ a_1 + \frac{h_1}{T} & a_0 + \frac{h_0}{T} & a_1 \\ a_1 + \frac{h_1}{T} & a_1 + \frac{h_1}{T} & a_0 \end{bmatrix}. \tag{4.75}$$

Note that $\mathbf{B}_{\text{face}}$ is neither circulant nor symmetric, and one diagonal element is always negative. Hence, in order for $\mathbf{B}_{\text{face}}$ to have an eigenvalue with a positive real part, h_0 cannot be less than what it was in the case of $\mathbf{B}_{\text{int}}$. Therefore, since eigenvalues are continuous functions of matrix elements, $\mathbf{B}_{\text{edge}}$ depicts the worst-case scenario.

Also observe that all the $\mathbf{B}$ matrices, representing the edges of the hypercube, have the same eigenvalues. This is a consequence of the fact that the $\mathbf{B}$ matrix that describes an edge can be obtained from the $\mathbf{B}$ matrix that describes any other edge by performing one row exchange and one column exchange. When the total number of row and column exchanges is even, the characteristic polynomial and, consequently, the eigenvalues of a matrix are preserved. Since all the edges of the hypercube are equivalent in this sense, any convenient choice of $\mathbf{B}_{\text{edge}}$, such as the one in (4.73), is appropriate.

The remainder of the proof employs Gerschgorin's circle theorems [49], [31], [50], [23]. One Gerschgorin circle of $\mathbf{B}_{\text{edge}}$ is centered at $a_0 + \dfrac{h_0}{T}$, with the radius $r_0 = \sum_{\vec{n} \neq \vec{0}} |a(\vec{n})|$. All the other Gerschgorin circles of $\mathbf{B}_{\text{edge}}$ are centered at a_0 with the maximum radius $r_{\text{sup}} = \left| a(\vec{k}) + \dfrac{h(\vec{k})}{T} \right| + \sum_{\vec{n} \neq \vec{0}, \vec{n} \neq \vec{k}} |a(\vec{n})|$. It is known that an isolated Gerschgorin circle contains exactly one eigenvalue [49], [50]. According to the hypothesis, (4.70), the circle centered at $a_0 + \dfrac{h_0}{T}$ is isolated from the rest and is situated entirely in the right half plane. Hence, $\exists i \ \mathrm{Re}\left(\lambda_i \left[\mathbf{B}_{\text{edge}}\right]\right) > 0$. Then by Lemma 4.5.1, edges of the hypercube do not contain stable fixed points. Therefore, $\vec{\chi} \in \{-1, 1\}^N$. ∎

Remark 4.5.3 *From (4.70), the* $\mathbf{B}_{\text{edge}}$ *matrix of the diagonal-state clipped M-lattice system is lower-triangular, and one of the eigenvalues is* $a_0 + \dfrac{h_0}{T}$. *Since the diagonal-state clipped M-lattice system is totally stable by Proposition 4.5.1, setting* $h_0 > -T a_0$ *guarantees that* $\vec{\chi} \in \{-1, 1\}^N$. *When* $T = 1$, *the diagonal-state clipped M-lattice system reduces to the reciprocal cellular neural network [42], [43], [44], [45], [46]). Thus, setting* $h_0 > -a_0$ *guarantees that* $\vec{\chi} \in \{-1, 1\}^N$.

It is interesting to study how the conditions of Proposition 4.5.2 vary in the limit as $T \to 0$. Intuitively, the specifications on h_0 ought to become less stringent in the high-gain limit, since any noise in the $\psi_i(t)$ (state) variables gets amplified to nearly ± 1 in the $\chi_i(t)$ (output) variables. In slightly more technical terms, a higher gain facilitates the process of making interior, face, and edge fixed points unstable. This alleviates from h_0 some of the burden of satisfying the conditions of Lemma 4.5.1.

Formally, letting $T \to 0$ in (4.70) gives:

$$\left\{ \begin{array}{rcl} h(\vec{0}) - \left| h(\vec{k}) \right| & > & 0, \\ h(\vec{0}) & > & 0, \\ \text{where } \vec{k} & = & \arg\max\limits_{\vec{n} \neq \vec{0}} |h(\vec{n})|. \end{array} \right. \tag{4.76}$$

Remark 4.5.4 *For the analog Hopfield network or the reciprocal cellular neural network operating in the high-gain limit, $h(\vec{0}) > 0$ guarantees that $\vec{\chi} \in \{-1, 1\}^N$.*

Remark 4.5.5 *Analysis of the general M-lattice system is also relevant for determining the stability of practical realizations of diagonal-state M-lattice systems (or analog Hopfield-type networks) [51], [52]. Since the circuit parasitics are unavoidable, the M-lattice system, (4.18), is a more accurate model of the actual electronic circuit than the diagonal-state M-lattice system, (4.37). While the diagonal elements of $\mathbf{A}$ reflect intended circuit parameters, the off-diagonal elements capture the parasitic resistances, coupling the state nodes of the network. Therefore, prior to committing the VLSI layout of, say, an analog Hopfield network to fabrication, the parasitics have to be carefully modeled and the entire design simulated as a full M-lattice system in order to verify that the network is totally stable and produces appropriate output values.*

4.6 Optimization With M-Lattice

The M-Lattice can solve non-linear programming problems for binary-valued and for real-valued non-linear objective functions. In either case, the constrained-optimization (*i.e.*, programming) problem is transformed into the more straightforward task of plain (unconstrained) optimization via the Karush-Kuhn-Tucker conditions [53].

4.6.1 Binary-Output Case

It has been shown that if $T \to 0$, then the analog Hopfield network seeks local optima on $\vec{\chi} \in \{-1, 1\}^N$ in the sense of the Hamming distance of one of objective functions that are multilinear polynomials [37], [35]. In this section, we state the conditions, under which the M-lattice system, (4.18) with $\vec{f}(\vec{\chi}(t)) \stackrel{\text{def}}{=} \vec{\nabla}_{\vec{\chi}}\Phi(\vec{\chi}(t))$, or the clipped M-lattice system, (4.48), do the same. The necessary mathematical background appears in Section B.4 of Appendix B.

Proposition 4.6.1 *Suppose that $\Phi(\vec{\chi})$ is a multilinear polynomial, let $T \to 0$, and assume that the M-lattice system is totally stable. Then the M-lattice system seeks local maxima of $\Phi(\vec{\chi})$ on $\vec{\chi} \in \{-1,1\}^N$ in the sense of the Hamming distance of one, provided that the following parity condition is satisfied:*

$$\forall\ i: \ \operatorname{sign}\left(-\sum_j a_{ij}\psi_j\right) \quad = \quad \operatorname{sign}(\psi_i)\,. \tag{4.77}$$

Proof: Due to the total stability, the M-lattice system converges in the limit as $t \to \infty$:

$$\begin{aligned} \vec{0} &= \mathbf{A}\vec{\psi} + \vec{\nabla}_{\vec{\chi}}\Phi(\vec{\chi}), \quad \text{or} & (4.78)\\ -\mathbf{A}\vec{\psi} &= \vec{\nabla}_{\vec{\chi}}\Phi(\vec{\chi}), \quad \text{or} & (4.79)\\ -\sum_j a_{ij}\psi_j &= \left[\vec{\nabla}_{\vec{\chi}}\Phi(\vec{\chi})\right]_i = \frac{\partial\Phi(\vec{\chi})}{\partial\chi_i}, \ \forall\ i. & (4.80) \end{aligned}$$

By Proposition B.4.2, $\vec{\chi}$ is a strict local maximum of Φ, provided that:

$$\forall\ i: \ \operatorname{sign}\left(\left[\vec{\nabla}_{\vec{\chi}}\Phi(\vec{\chi})\right]_i\right) \quad = \quad \operatorname{sign}(\chi_i)\,. \tag{4.81}$$

But $\operatorname{sign}(\chi_i) = \operatorname{sign}(\psi_i)$ for all standard sigmoidal (refer to (4.2), (4.3), and Assumption 4.2.1) and clipping (refer to (4.47) as one example) warping functions. With that in mind, inserting (4.81) into (4.80) recovers the conclusion, (4.77). ∎

4.6.2 Real-Output Case

If $\vec{\chi}(t) = \frac{1}{T}\vec{\psi}(t)$ and Assumption 4.2.3 is satisfied, then the non-clipped version of (4.48) becomes a gradient system, associated with the following auxiliary function:

$$E(t) \stackrel{\text{def}}{=} E(\vec{\psi}(t)) \quad = \quad \frac{1}{2}\vec{\psi}(t)^T\mathbf{A}\vec{\psi}(t) + T\Phi(\vec{\chi}(t)). \tag{4.82}$$

This is confirmed by taking the gradient of $E(\vec{\psi}(t))$ with respect to $\vec{\psi}(t)$:

$$\vec{\nabla}_{\vec{\psi}} E(\vec{\psi}(t)) \;=\; \mathbf{A}\vec{\psi}(t) + T\left(\frac{1}{T}\mathbf{I}\right)\vec{\nabla}_{\vec{\chi}}\Phi(\vec{\chi}(t)) \;=\; \mathbf{A}\vec{\psi}(t) + \vec{\nabla}_{\vec{\chi}}\Phi(\vec{\chi}(t)), \tag{4.83}$$

which vanishes at the fixed points. Since Assumption 4.2.3 is satisfied, Lemma 4.3.3 still holds, and by Lemma B.1.1, (4.48) does not contain cycles. Then it follows that (4.48) is totally stable by Lemma 4.3.2, Lemma 4.3.4, and Proposition 4.3.2. Hence, expressing $\vec{\nabla}_{\vec{\chi}}\Phi(\vec{\chi})$ in terms of $\vec{\nabla}_{\vec{\psi}} E(\vec{\psi}(t))$ and inserting the result into (4.48) leads to fixed points, $\vec{\psi}$, corresponding to local maxima of $E(\vec{\psi}(t))$ in (4.82).

We summarize the result for $T = 1$.

Proposition 4.6.2 *Suppose that Assumption 4.2.3 is satisfied with $\vec{\chi}(t) = \vec{\psi}(t)$ and set $\vec{\nabla}_{\vec{\chi}}\Phi(\vec{\chi}(t))$ to:*

$$\vec{\nabla}_{\vec{\chi}}\Phi(\vec{\chi}(t)) \;=\; \vec{\nabla}_{\vec{\psi}} E(\vec{\psi}(t)) - \mathbf{A}\vec{\psi}(t). \tag{4.84}$$

Then the M-lattice system, (4.48), seeks local maxima of the following function:

$$E(\vec{\psi}) \;=\; \frac{1}{2}\vec{\psi}^T\mathbf{A}\vec{\psi} + \Phi(\vec{\chi}) \tag{4.85}$$

with respect to $\vec{\psi}$.

Remark 4.6.1 *If $\mathbf{A}$ is set to $\mathbf{0}$, then (4.48) turns into the non-linear programming neural network [39], [40], [41]. When $\mathbf{A} = \mathbf{0}$, Lemma 4.3.3 does not apply. Thus, in order to operate this system, one must assume that $\vec{\psi}(t)$ is bounded. This is a stronger assumption than Assumption 4.2.3, required by Proposition 4.6.2.*

4.7 Pattern-Forming Property Of M-Lattice

When the M-lattice system is organized as shown in Figure 4.4(c) and is the discrete-space version of a particular case of the general reaction-diffusion PDE, stated in (2.3), its small-signal equivalent becomes a convolutionally-coupled M-lattice system. In such cases, determining whether or not the original M-lattice system is capable of pattern formation is particularly straightforward. The given M-lattice system possesses the pattern-forming property if its corresponding small-signal dynamical system is a pattern-forming convolutionally-coupled M-lattice system.

4.8 Chapter Summary

We developed and mathematically analyzed the M-lattice, proposed as a novel alternative to reaction-diffusion. While retaining the crucial pattern-forming property of reaction-diffusion, the M-lattice is bounded. In addition, it allows more flexible interactions among its variables than its close relatives, such as the analog Hopfield network and the cellular neural network. Flexibility and boundedness enable the M-lattice to capture the behavior of reaction-diffusion and many physical systems in a controlled manner.

Investigating the convergence properties of various forms of the M-lattice and – in the cases of convergence – studying the types of equilibrium output values has been this work's main theoretical thrust. For a subclass of the M-lattice, called the diagonal-state clipped M-lattice system, we provide a proof of the total stability, and this proof applies to the cellular neural network as well. The total stability is an important property, because it guarantees convergence to equilibrium states with potentially practical interpretation.

By analyzing the stability of fixed points, we derive the conditions for driving the equilibrium outputs of the M-lattice to binary values. Systems capable of converging to binary-valued equilibria are useful for solving coordinated decisions problems, and this binarization result also contributes to the theory of other related

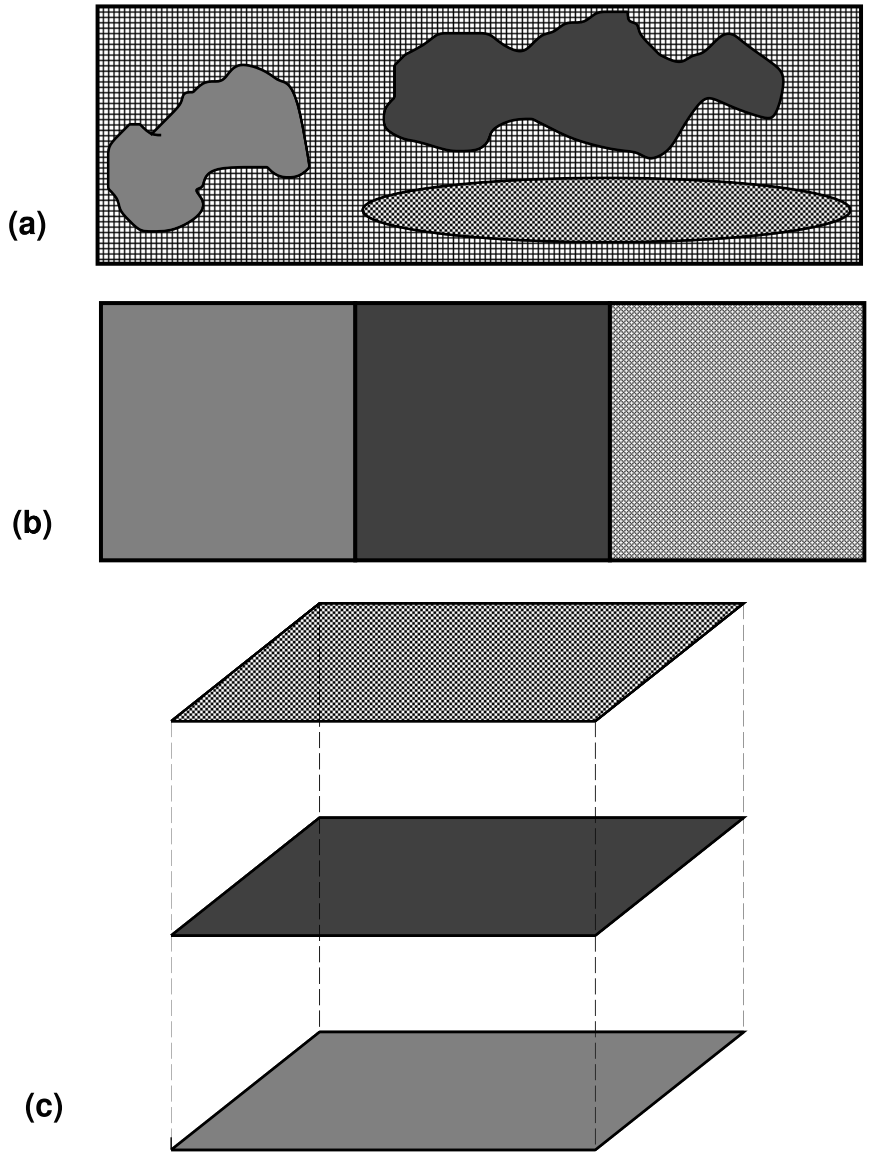


Figure 4.4: Three of many ways of arranging the M-lattice on a spatial grid. (a) layers with flexibly defined boundaries with arbitrary linear and non-linear intra-layer and inter-layer interactions; (b) rectangular layers with arbitrary linear and non-linear intra-layer and inter-layer interactions; (c) rectangular layers with the intra-layer interactions restricted to be linear and the inter-layer interactions restricted to involve only the output variables corresponding to the same spatial index (*i.e.*, at vertically aligned sites).

models. Specifically, for the cellular neural network, the analysis in this chapter is a precise formulation of an earlier argument based on circuit diagrams. And for certain special cases of the analog Hopfield network, this analysis explains why the output variables converge to binary values even with non-zero neuron auto-connections. This behavior, observed in computer simulation by other researchers for quite some time, has been unexplained until now.

As another practical interpretation of equilibrium states, we have determined the conditions under which various forms of the M-lattice converge to local optima of non-linear objective functions. For binary-valued outputs, the M-lattice, like the analog Hopfield network, is capable of seeking local optima of multilinear polynomials in the sense of the Hamming distance of one. For real-valued outputs, the clipped M-lattice performs optimization in the standard sense and provides a realistic mathematical model for the non-linear programming neural network circuit.

This chapter ends with the observation that, based on the definition given earlier, many types of the M-lattice possess the pattern-forming property. This property makes the M-lattice suitable for pattern-synthesis and pattern-extraction applications.

Table 4.1 compares the new M-lattice model with its predecessors.

Model	Linear Interactions (On State Variables)	Sigmoid	Non-Linear Interactions (On Output Variables)	External Input Inclusion
RD	diffusion only	not required	limited variety	wide variety
Hopfield Network	diagonal matrix*	required	symmetric matrix	diagonal matrix
Non-Linear Programming Circuit	diagonal matrix*	not required	limited variety	any
Cellular Neural Network	diagonal matrix*	required	symmetric matrix	matrix
M-Lattice	matrix*	required	wide variety	any

Table 4.1: Comparison of the M-lattice system with other related models. The "*" indicates that all eigenvalues must have a negative real part. The column titled "External Input Inclusion" refers to the variety of ways the input signals can intertwine before the result entering the system.

Chapter 5

Applications

This chapter demonstrates how various modes of operation of the M-lattice system, presented in Chapter 4, are applied to image and sound synthesis, image restoration and enhancement, image halftoning, and non-linear programming.

5.1 Synthesizing Visual And Sound Textures

As a "sanity-check" application of the new M-lattice, we first use it to implement Turing's reaction-diffusion system. We verify that the M-lattice can simulate the basic reaction-diffusion mechanism by duplicating a number of published visual textures. Then we generate new visual and sound textures.

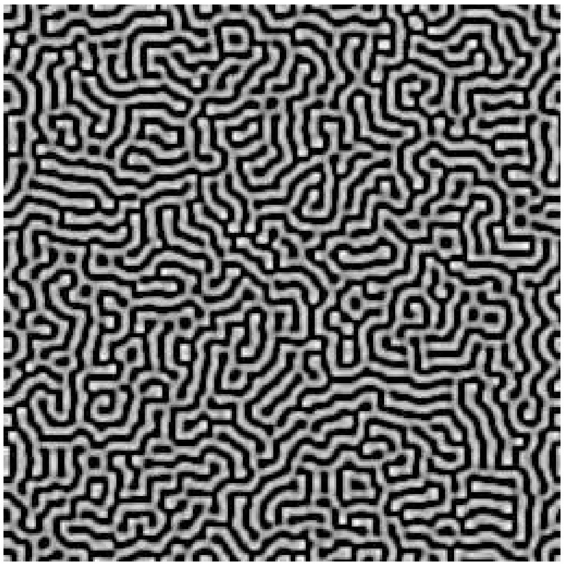
5.1.1 M-Lattice For Synthesizing Visual Textures With Turing's Reaction-Diffusion

Consider the following modification of Turing's two-morphogen reaction-diffusion system (in 1-D for the simplicity of notation):

$$\begin{aligned}
\frac{d\psi_A(n_x,t)}{dt} &= \chi_A(n_x,t)\cdot\chi_I(n_x,t) - \psi_A(n_x,t) - 12 \\
&+ D_A[\psi_A(n_x+1,t) - 2\psi_A(n_x,t) + \psi_A(n_x-1,t)], \\
\frac{d\psi_I(n_x,t)}{dt} &= 16 - \chi_A(n_x,t)\cdot\chi_I(n_x,t) \\
&+ D_I[\psi_I(n_x+1,t) - 2\psi_I(n_x,t) + \psi_I(n_x-1,t)],
\end{aligned} \tag{5.1}$$

where $G_i(u)$ uses (4.47) with $T = 1$ and is clipped below at zero. Clearly, by collecting the linear and the non-linear terms, (5.1) can be phrased in the form of (4.5). The basic linear term is the discretized diffusion. The linear term of the activator also absorbs $-\psi_A(n_x,t)$, originally a part of reaction. With this arrangement of terms, (5.1) satisfies Assumption 4.3.1. The non-linear term is comprised of the non-linear part of reaction, whose functional specification is the same for all spatial indices, n_x, and the actual reaction interaction is only between the site itself and the site at the same spatial index on the other lattice.

In order for (5.1) to form patterns, it must obey the local stability / instability requirements of Definition 2.5.1. Since (4.47) with $T = 1$ is a unity-slope straight line near the fixed point, $\psi_A(n_x, t = t_0) = \psi_I(n_x, t = t_0) = 4$, this M-lattice system inherits all the pattern-forming properties of Turing's reaction-diffusion system. In fact, we have found that since the sigmoid with $T = 1$ also appears as a unity-slope straight line for small deviations about the fixed point, this M-lattice system retains all the pattern-forming properties of Turing's reaction-diffusion system if $G_i(u)$ uses the sigmoid of (4.2) with $T = 1$ and is saturated below at zero.



(a)

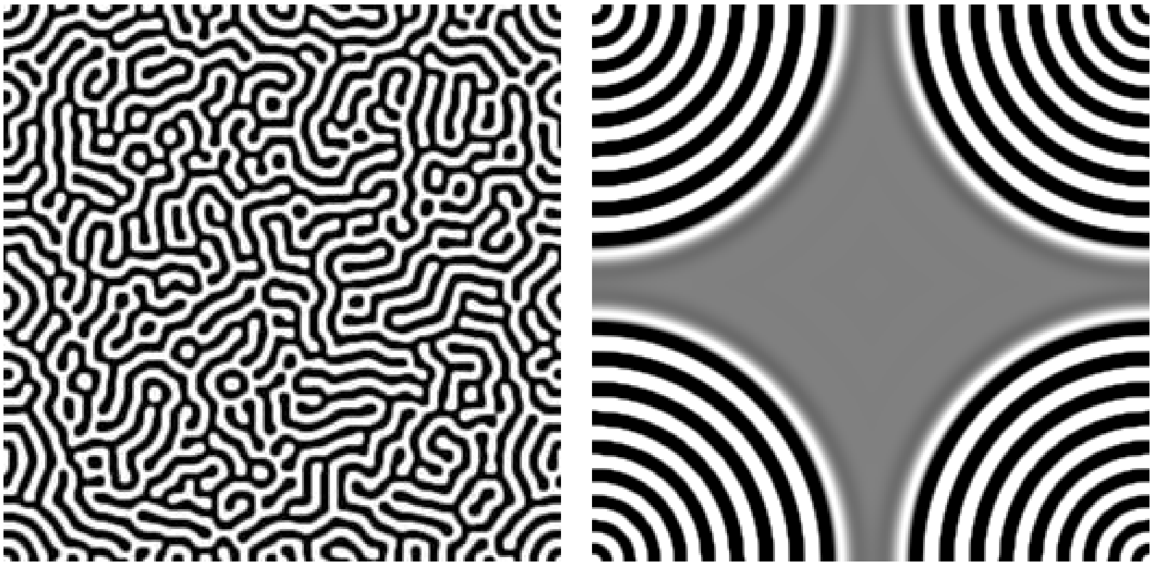
(b)

Simulation of this system gives results that were very close to those of the original Turing system with the clipping done inside the numerical integration method. This is not surprising, because the M-lattice system takes care of the clipping by the virtue of its form.

In addition to repeating the spots of Figure 2.4 and the stripes of Figure 2.5, a number of additional textures have been synthesized with the M-lattice, operating in the reaction-diffusion mode. This "texture gallery" appears in Figure 5.1.

5.1.2 M-Lattice For Synthesizing Sound Textures

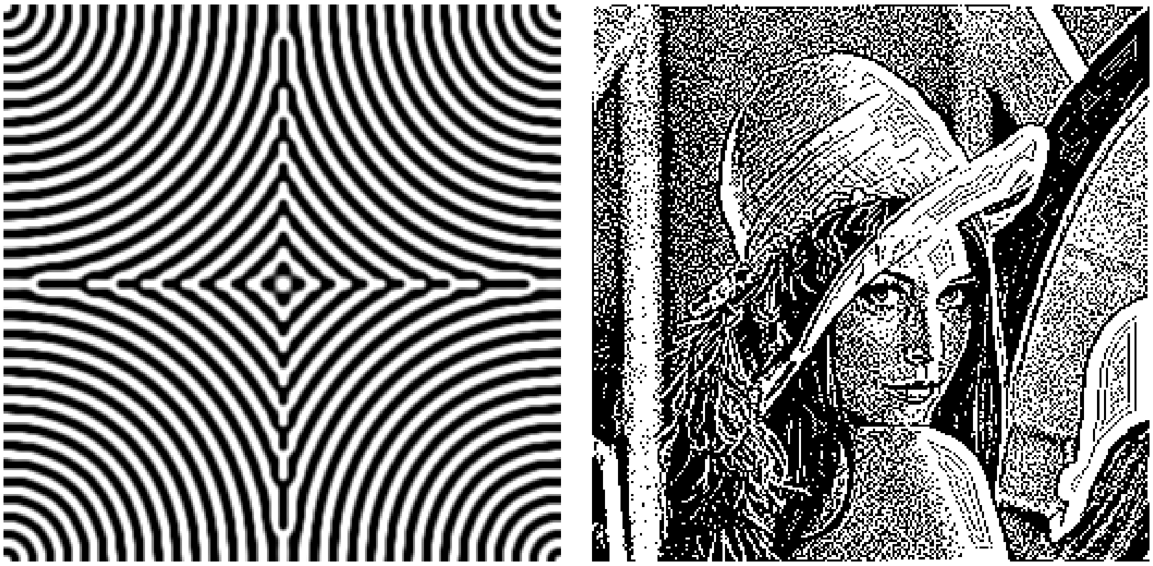
We have seen that reaction-diffusion on a 2-D spatial grid can synthesize a wide variety of natural-looking visual textures. Suppose that we remove one dimension. Then one can design the resulting 1-D reaction-diffusion system to synthesize natural 1-D patterns. Treating the hydranth as the ring of cells proved useful for trying to explain the growth of tentacles in tubularia (see Section 2.4.4 of Chapter 2) [1].



(c) (d)

This model can also be applied to sound synthesis if the audio waveform is treated as the ring of sound samples. The goal is to harness the pattern-forming property of reaction-diffusion to synthesize natural-sounding music textures. The bounded and flexible M-lattice facilitates the design process, captured in Figure 5.2. For the purposes of listening, the final sample values are sequenced and played through the loudspeaker. In fact, the periodic boundary conditions allow the "seemless tiling" (repetition) of the same sequence. Thus, the sound waveform can be looped over and over for prolonged listening without audible discontinuities.

The present technique belongs to the class of "tape methods" (as opposed to "real-time methods") of sound synthesis. This is because all samples are updated at the same time during synthesis, and the temporal order is not enforced until playback.



(e) (f)

Figure 5.1: Reaction-diffusion texture gallery, synthesized by the M-lattice. (a) the "monkey brain" texture, generated by using Turing's model with $D_I = 256$ and the evocator having a random spatial-frequency spectrum; (b) the "worms" texture, generated by using Turing's model with $D_I = 256$ and the evocator having random spatial-domain samples; (c) the "wiggles" texture, generated by using a linear-reaction reaction-diffusion system with $D_I = 256$ and the evocator having a random spatial-frequency spectrum; (d) the "circles" texture, generated by using a linear-reaction reaction-diffusion system with $D_I = 400$ and the evocator having a white spatial-frequency spectrum; (e) the "target" texture, generated by using a linear-reaction reaction-diffusion system with $D_I = 256$ and the evocator having a white spatial-frequency spectrum; (f) the "artistically-halftoned Lena" image, generated by using a linear-reaction reaction-diffusion system with $D_I = 400$, the original "Lena" image of Figure 5.8(a) as the evocator, and the system designed as an aggressive band-pass filter.

By experimenting with a large number of linear and non-linear reaction functions, describing the correlations among samples on different scales, we have synthesized several natural sounds, most notably:

- burning match;
- cello; and
- brass.

These sounds possess distinctively rich timbre. Furthermore, it has been remarked that they exhibit "warm" analog-like qualities, even though the synthesis algorithms have been implemented on the digital computer [54]. We attribute this phenomenon to the fact that the M-lattice is an analog system, and its evolution is tracked accurately in simulation by discretizing the time axis into finely-quantized steps.

5.2 Estimating Local Orientation

For the applications that follow in Section 5.3 and Section 5.4.1, we will use the local orientation in the image to guide the action of the M-lattice system. The physiological evidence for orientation detectors in the human visual system system [55] has led to widespread recognition of the importance of orientation for both human vision and perceptually-based image processing. Not only is orientation perceptually significant, but image processing algorithms that sense local orientation are better equipped to adapt to inhomogeneous data. Rather than compute orientation at all possible angles and then decide which angle dominates, we employ the computation-saving "steerable" set of basis filters described in [56] and used in [57]. The output of the

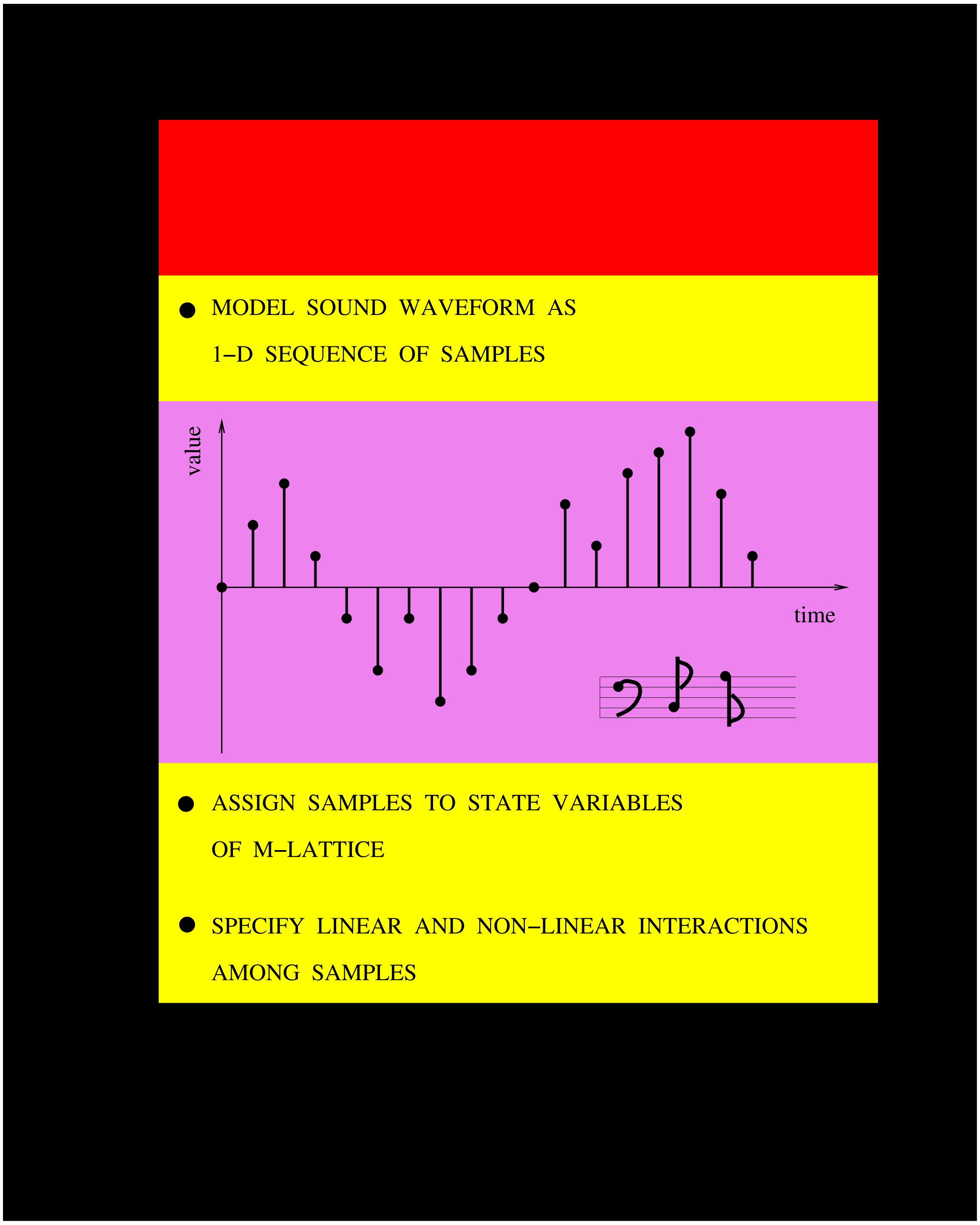


Figure 5.2: Synthesizing reaction-diffusion sound textures using the M-Lattice.

steerable filters at each pixel gives the angle, $\theta_i \in [-\pi, \pi]$, and relative strength (or magnitude), $m_i \in [0, 1]$, of the dominant orientation present at that pixel.

For example, to design a low-pass adaptive filter that rotates to the dominant orientation, denote the diagonal matrix of variances by $\mathbf{V}_i$ and the rotation matrix by $\mathbf{\Theta}_i$:

$$\mathbf{V}_i = \begin{bmatrix} \sigma_{i,x}^2 & 0 \\ 0 & \sigma_{i,y}^2 \end{bmatrix}, \quad \mathbf{\Theta}_i = \begin{bmatrix} \cos\theta_i & -\sin\theta_i \\ \sin\theta_i & \cos\theta_i \end{bmatrix}. \tag{5.2}$$

The relative sizes of $\sigma_{i,x}^2$ and $\sigma_{i,y}^2$ depend on m_i and determine the skewness of filters with respect to the dominant orientation:

$$\sigma_{i,y}^2 = \frac{L}{2}(1 - m_i), \quad \sigma_{i,x}^2 = L - \sigma_{i,y}^2, \tag{5.3}$$

where $L \times L$ is the size of the filter mask in pixels. Let $\vec{n} \in \mathcal{Z}^2$ be the pixel position. Then the (unnormalized) oriented low-pass filter is given by:

$$h_i(\vec{n}) \;=\; \exp\left\{-\vec{n}^T \mathbf{\Theta}_i^T \mathbf{V}_i \mathbf{\Theta}_i \vec{n}\right\}. \tag{5.4}$$

5.3 Restoration And Halftoning Of Fingerprints Using M-Lattice System

A typical fingerprint identification system contains a pre-processing step, which involves the halftoning of the original scanned and finely-quantized fingerprint image. The essential steps comprising the identification sequence are: determining the type of the fingerprint, counting of ridges and bifurcations, and locating the core. A binary fingerprint image is more amenable for these tasks than a gray-scale fingerprint image [58].

In this section, we propose a pre-processing scheme that not only halftones the original fingerprint image, but also removes artifacts that can hinder the identification process. The method uses the ability of the clipped M-lattice system to excite locally-growing stationary spatial waves, which are the signature of reaction-diffusion systems, as well as to produce equilibrium images that have binary pixel values.

The motivation for using the reaction-diffusion aspect of the clipped M-lattice system is that fingerprint images have distinct patterns of thin curves, remotely resembling zebra stripes. Reinforcing the harmonics that create these curves will emphasize the essentials of the fingerprint, while suppressing the artifacts [59].

In light of this analogy, it is interesting to note that the reaction-diffusion approach to image restoration models the signal (as opposed to modeling the noise [60]). Many traditional, such as Bayesian or Wiener, approaches model the degradation and then design the model parameters so as to attenuate it [61], [62], [63], [25], [64], [65].

As shown in Chapter 4, halftoned images that are fixed points of the M-lattice system are asymptotically stable. This means that even though reaction-diffusion is a small-signal phenomenon, the large-signal evolution of the system toward a binary output image does not destroy the restoration performed by the linear behavior.

5.3.1 Small-Signal Regime: Reaction-Diffusion

Following the discussion of Section 4.5.2 of Chapter 4, let $\mathbf{A}$ and $\mathbf{H}$ be block-circulant symmetric matrices. Then the clipped M-lattice system, (4.48), can be written as follows:

$$\frac{d\psi(\vec{n},t)}{dt} = a(\vec{n}) * \psi(\vec{n},t) + s(\vec{n}) - h(\vec{n}) * \chi(\vec{n},t), \tag{5.5}$$

where $\vec{n} \in \mathcal{Z}^2$, $a(\vec{n})$ and $h(\vec{n})$ are the FIR filters, corresponding to $\mathbf{A}$ and $\mathbf{H}$, respectively, and $s(\vec{n}) \in [-1,1]$ is the original finely-quantized input image signal [1]. The advantage of using this type of the clipped M-lattice system for the pre-processing of fingerprints is that it can be guaranteed to produce binary outputs.

Choose $\mathbf{A}$ and $\mathbf{H}$ such that the unique interior fixed point, $\psi(\vec{n}) \in (-1,1)$, of (5.5) is at $s(\vec{n})$. Denote the DFT representations of the filters by $A(\vec{k})$ and $H(\vec{k})$. Then all the $A(\vec{k})$ coefficients must be negative [23]. Before $\psi(\vec{n},t)$ reaches the clipping levels of (4.47), (5.5) simplifies to:

$$\frac{d\psi(\vec{n},t)}{dt} = s(\vec{n}) + \left(a(\vec{n}) - \frac{1}{T}h(\vec{n})\right) * \psi(\vec{n},t). \tag{5.6}$$

This is a one-morphogen reaction-diffusion system, introduced in Section 3.2.4 of Chapter 3. Taking the DFT of both sides of (5.6) yields:

$$\frac{d\psi(\vec{k},t)}{dt} = S(\vec{k}) + \left(A(\vec{k}) - \frac{1}{T}H(\vec{k})\right)\psi(\vec{k},t), \tag{5.7}$$

whose solution for each $\vec{k}$ is:

$$\psi(\vec{k},t) = \left[S(\vec{k}) + \frac{S(\vec{k})}{F(\vec{k})}\right]\exp\left\{F(\vec{k})t\right\} - \frac{S(\vec{k})}{F(\vec{k})}, \tag{5.8}$$

where $F(\vec{k}) \overset{\text{def}}{=} A(\vec{k}) - \frac{1}{T}H(\vec{k})$, and the initial condition is set to $S(\vec{k})$, the DFT equivalent of the original image.

In Chapter 2 and Chapter 3, it was shown that making $F(\vec{k})$ positive for a set of spatial frequencies creates the onset of growing spatially-stationary waves.

The $H(\vec{k})$ filter is designed in a way that incorporates the orientation information of the fingerprint image as described in Section 5.2. Pertinent to fingerprint restoration is the kind of filtering that delineates the ridges, while canceling fluctuations in the DC level and getting rid of extraneous information. Thus, $H(\vec{k})$ and $A(\vec{k})$ are constructed in a way that makes the frequency bands corresponding to the ridges have negative DFT coefficients, and the $\frac{1}{T}$ factor amplifies the effect.

[1] Actually, $s(\vec{n})$ is obtained by scaling and shifting the original 256-gray-level image, $I(\vec{n})$: $s(\vec{n}) = \left(\frac{I(\vec{n})}{128}\right) - 1$.

5.3.2 Large-Signal Regime: Halftoning

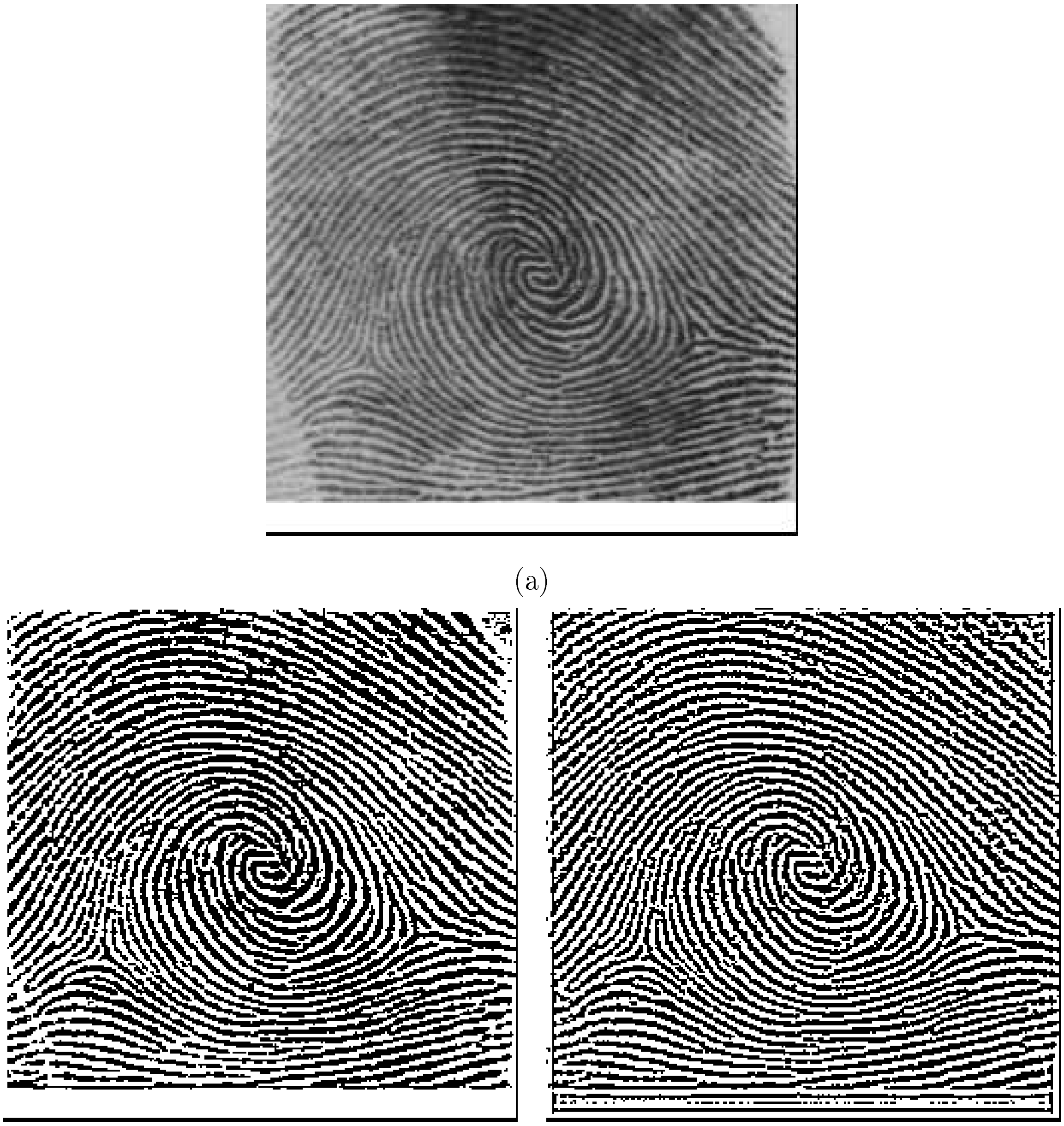
Proposition 4.5.2 gives sufficient conditions for ensuring that the large-signal equilibrium output pattern is binary. As was discussed in Section 4.5.2 of Chapter 4, these conditions place restrictions on the elements of $h(\vec{n})$. The needed additional flexibility is provided by the $A(\vec{k})$ filter, which has negative DFT coefficients. For instance, in fingerprint restoration, we use $A(\vec{k})$ to cancel the unwanted harmonics, most importantly the DC term, $\vec{k} = (\vec{0})$, which is not possible with $H(\vec{k})$ acting alone or with $a(\vec{n}) \propto -\delta(\vec{n})$.

Figure 5.3(a) is a typical scanned and finely-quantized fingerprint image from the NIST database. The original image is 512×512 pixels and was low-pass filtered and down-sampled by a factor of 2 in each dimension in order to speed up the computation. From the figure, it can be seen that the original fingerprint is corrupted by a number of scratches, and several regions are obscured by uneven illumination. As shown in Figure 5.3(b), the standard fingerprint halftoning method, based on adaptive median filtering and thresholding, only makes these artifacts more apparent, because it increases the image's contrast [66]. The adaptive threshold is set to the average of the minimum and the maximum gray levels within some neighborhood surrounding each pixel of the original fingerprint image. The optimal size of the window was determined to be 5×5 pixels by trial and error. Other standard halftoning methods, such as ordered dither or error diffusion, will perform poorly also, because they have no built-in restoration mechanism and will halftone both signal and noise alike.

Using a two-stage system, consisting of some conventional image restoration algorithm, followed by adaptive-thresholding type of halftoning is another viable alternative. However, there are two arguments in favor of using the clipped M-lattice system-based approach. First, the analysis of Chapter 3 implies that the attainable signal-to-noise ratios can be very large. Essentially, the clipped M-lattice system applies the filters, $a(\vec{n})$ and $h(\vec{n})$, an infinite number of times by the virtue of being a continuous-time system. Second, no separate halftoning step is needed, since the clipped M-lattice system binarizes the image by setting the elements of $h(\vec{n})$ in accordance with Proposition 4.5.2.

Other researchers have succeeded in using the reaction-diffusion paradigm for performing operations that are relevant to the pre-processing of fingerprint images. One effort, which, incidentally, was not specifically targeted for fingerprint restoration, deals with replacing portions of fingerprints with patches, whose visual appearance resembles that of a generic fingerprint texture [67], [68]. However, the inserted patch may turn out to be substantially dissimilar from the missing section of the original image. Another report describes the restoration of noisy fingerprints with a particular reaction-diffusion system [29]. However, the output image is not a halftone. Our new technique is inspired by these previous approaches and accomplishes restoration and halftoning simultaneously.

Figure 5.3(c) displays the processed fingerprint image. The scratches have been removed and the unevennesses in the DC levels throughout the image have been eliminated. Essential detail such as ridges and bifurcations appear as continuous black curves, distinctly enhanced against a noise-free white background. Moreover, ridges and bifurcations have been extended even into the regions where they are barely detectable in the original image. This illustrates the celebrated synergetic property of reaction-diffusion systems: the emergence from noise of a spatial pattern, whose qualitative characteristics are pre-determined by the system's parameters [1], [12]. Using the connection machine (CM-2), the final image is produced in 25 iterations at the time step of 0.1 sec. For viewing convenience, Figure 5.4 shows a 128×128 pixel middle-top section of each respective image of Figure 5.3, magnified by a factor of 2 in both dimensions.



(a)

(b) (c)

Figure 5.3: Restoration and halftoning of fingerprints. (a) the original "fingerprint" image; (b) the "fingerprint" image halftoned by a standard adaptive-threshold method; (c) the "fingerprint" image restored and halftoned by the clipped M-lattice system operating in the reaction-diffusion mode utilizing orientation information at each pixel of the original.

Figure 5.4: Magnification of a 128×128 pixel middle-top section of the images in Figure 5.3. (a) original; (b) halftoned by a standard adaptive-threshold method; (c) restored and halftoned by the clipped M-lattice system.

5.3.3 Reaction-Diffusion And Optimization

In addition to acting as a reaction-diffusion system in the small-signal or linear mode of operation, (5.5) is capable of performing non-linear optimization in the large-signal mode. Following the discussion of Section 4.6, suppose that $h(\vec{n})$, designed to amplify the required spatial harmonics, allows setting $h(\vec{0}) = 0$. Also, let $T \rightarrow 0$, since this is the only means of obtaining binary outputs. Finally, ensure that $a(\vec{n}) * \psi(\vec{n})$ and $\psi(\vec{n})$ have the opposite signs $\forall\ \vec{n}$. Then the resulting equilibrium image is a local minimum (in the sense of the Hamming distance of one) of the following objective function:

$$-\Phi(\vec{\chi}) \quad = \quad -\left(\vec{s}^T\vec{\chi} - \vec{\chi}^T\mathbf{H}\vec{\chi}\right), \tag{5.9}$$

where $\mathbf{H}$ is the equivalent matrix representation of $h(\vec{n})$.

5.3.4 Comparison To Related Models

The cellular neural network can also give rise to reaction-diffusion by designing the $h(\vec{n})$ filter as discussed in Section 5.2. However, the cellular neural network does not perform optimization in the sense of the Hamming distance of one, because it requires $h(\vec{0}) \neq 0$ in order to assure binary outputs [42].

The analog Hopfield network performs optimization in the sense of the Hamming distance of one, because it has $h(\vec{0}) = 0$ [37], [35]. But the $a(\vec{n})$ filter and the $h(\vec{n})$ filter are undefined, unless they are shift invariant, in which case the analog Hopfield network becomes a cellular neural network. Furthermore, when the $a(\vec{n})$ filter is defined, $a(\vec{n}) \propto -\delta(\vec{n})$, so that $A(\vec{k})$ is the same negative number for all $(\vec{k})$. This makes it difficult, if not impossible, to suppress the growth of the unwanted harmonics.

In contrast, the M-lattice system combines the attractive features of both networks. The flexibility brought about by the $a(\vec{n})$ filter makes it easier to design the frequency bands that are amplified during the small-signal reaction-diffusion regime. Finally, if the $\mathbf{A}$, which corresponds to the $a(\vec{n})$ filter is substantially diagonally dominant, then the M-lattice system also performs optimization in the sense of the Hamming distance of one, since $h(\vec{0}) = 0$.

5.4 Halftoning

The halftone process, in which the tonal values of the image are represented by dots of varying sizes on a white background has played an important role in the graphic arts and printing industries since the late 1800s [69].

Halftoning can be viewed as a transformation from the gray-scale domain to the dot-matrix domain. Because of the limited spatial frequency response of the human visual system, the dots appear to merge at normal viewing distances (*e.g.*, 8 – 14 inches), thereby producing the same impression as the original gray-scale image. Since the final part of the overall display system is the human viewer, the halftoning method must perceptually yield an image which appears similar to the original gray-scale image [70].

The recent proliferation of medium-resolution (*e.g.*, 300 $\frac{\text{dots}}{\text{inch}}$) binary output devices, such as FAX machines and laser printers has brought increased attention to optimal (in the sense of the eye and the printer models) image halftoning methods that meet inexpensive parallel hardware constraints [71]. Halftoning methods that are optimal in the least-squares sense are summarized in Appendix E.

5.4.1 Halftoning Using Diagonal-State M-Lattice

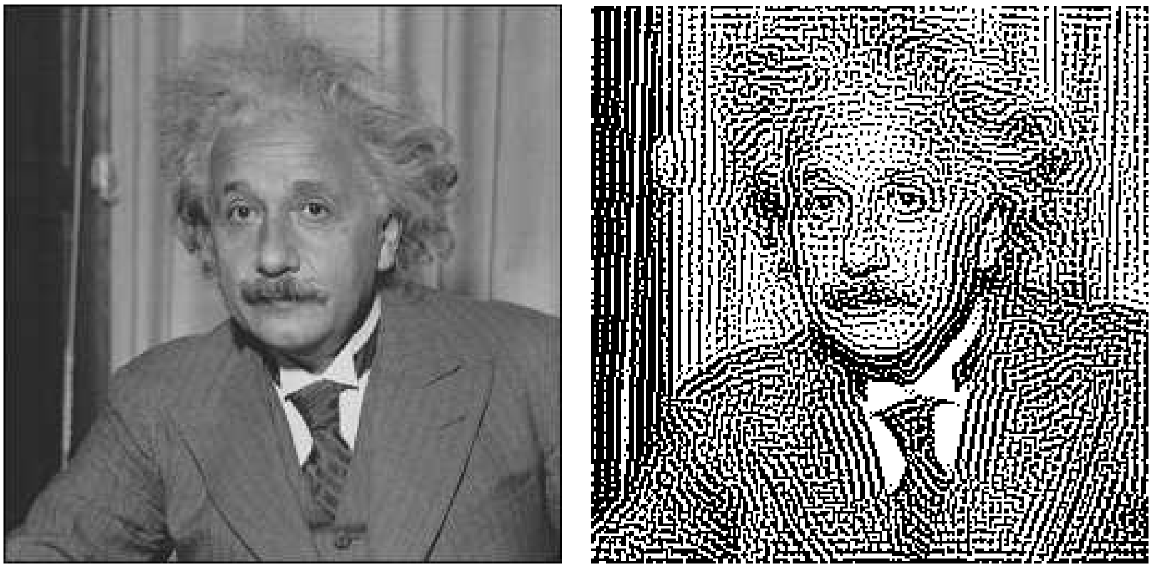
In this section, we consider the problem of synthesizing a binary caricature that brings out the directional content of an image. The resulting halftoning method must yield an image that appears similar to the original gray-scale image in some indirect sense. A least-squares halftoning approach, such as the filtered-squared-error halftoning method of Section E.1, is appropriate for this task, because it can employ an explicit model of perception as the measure of performance [72]. Here we show how to implement such an approach using the diagonal-state M-lattice system.

Suppose $\vec{n} \in \mathcal{Z}^2$; $s(\vec{n}) \in [-1, 1]$ is the finely-quantized original input image signal; $y(\vec{n}) \in \{-1, 1\}$ is the output halftone image; and $h(\vec{n})$ is a 2-D filter (not necessarily the same as $h(\vec{n})$ in the previous section). Let $\mathbf{B} = \mathbf{H}^T\mathbf{H}$, where $\mathbf{H}$ is a circulant matrix with $h(\vec{n})$ in the first row. The problem of halftoning can be stated as a non-linear program. In order to solve this problem using the diagonal-state M-lattice system, we combine the objective function to be minimized, (E.6), with the N constraints, (E.7), into the Lagrangian cost functional with the help of the Karush-Kuhn-Tucker conditions [53]:

$$
\begin{aligned}
& \min_{\vec{y}} \mathcal{L}(\vec{y}), \quad \text{where} \\
\mathcal{L}(\vec{y}) \;&=\; \frac{1}{2}\vec{y}^T\mathbf{B}\vec{y} - (\mathbf{B}\vec{s})^T\vec{y} + \frac{1}{2}\sum_i p_i(y_i^2 - 1), \qquad (5.10) \\
p_i \;&\leq\; 0, \quad p_i(y_i^2 - 1) = 0. \qquad (5.11)
\end{aligned}
$$

The Lagrange multipliers, p_i, are the varying penalty terms that enforce the constraints according to (5.11). As a result, the unconstrained minimization of $\mathcal{L}(\vec{y})$ in (5.10) produces the optimal halftone image.

The optimization problem, (5.10), is "programmed" onto the M-lattice system, (4.48), by setting $\vec{y}$ equal to $\vec{\chi}$, $\Phi(\vec{\chi})$ to $-\mathcal{L}(\vec{y})$, and taking partial derivatives. This



(a) (b)

Figure 5.5: Orientation-sensitive halftoning. (a) the original "Einstein" image; (b) the "Einstein" image adaptively halftoned using orientation information at each pixel of the original.

yields:

$$\frac{d\vec{\psi}(t)}{dt} = \mathbf{A}\vec{\psi}(t) + \mathbf{B}\vec{s} - \mathbf{B}\vec{\chi}(t) - \mathbf{P}\vec{\chi}(t), \tag{5.12}$$

where $\mathbf{P} = \text{Diag}\ \{p_1,\ \ldots,\ p_N\}$. The elements of $a(\vec{n})$ are chosen so as to guide the system towards an optimum corresponding to a perceptually-pleasant halftone. In the next section, we will show that $\mathbf{A} = \mathbf{B} - \mathbf{I}$ is a good choice, because it filters out objectionable correlated spatial patterns.

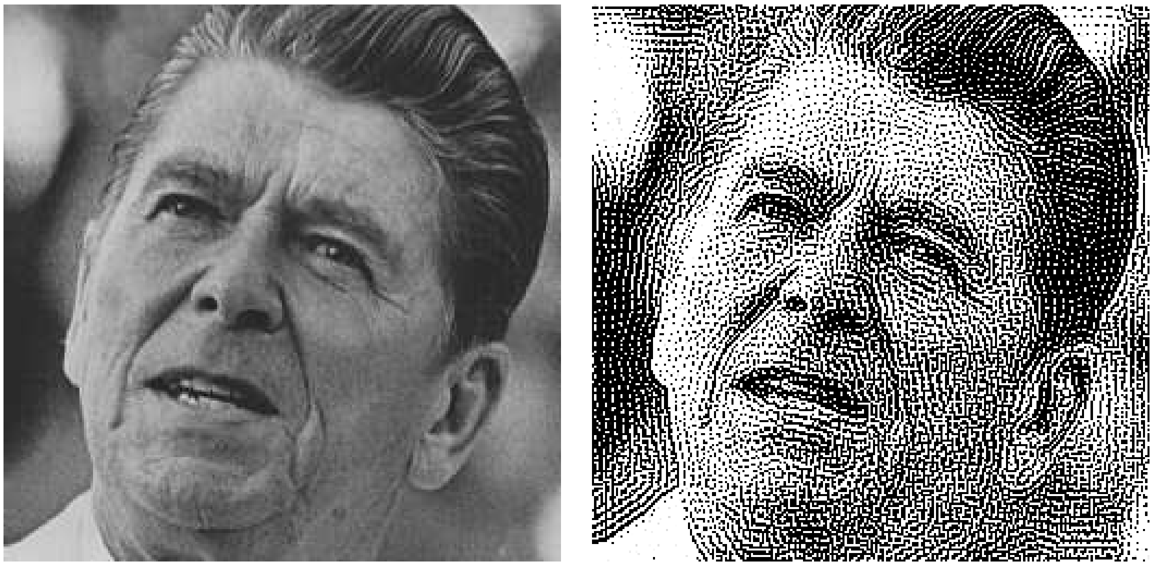
Halftoning with the Hopfield network would be similar, but requires setting $b_{ii} \geq 0$ [73]. Otherwise, the optimal values of y_i will not be binary [37]. In addition, the Hopfield network restricts the quality metric to be a symmetric matrix, $\mathbf{B}$. Since no effort is made here to design $\mathbf{H}$ in a way that would result in $b_{ii} \geq 0$, the non-linear constraints provide the only mechanism for driving the output pixels to the limits of the gray scale.

Treating halftoning as a non-linear programming problem and solving it with the diagonal-state M-lattice system offers considerable flexibility in the choice of the quality metric and in the functional form of constraints. In order to demonstrate this flexibility, we incorporated orientation detection (see Section 5.2) into the halftoning quality metric. The adaptive filter matrix, $\mathbf{H}$, was designed so as to include the information about the dominant orientation at each pixel of the original image, shown in Figure 5.5(a). The corresponding binary image, produced by the diagonal-state M-lattice system, appears in Figure 5.5(b). Since no effort is made to design $\mathbf{H}$ in a way that would result in $b_{ii} \geq 0$, the non-linear constraints provide the only mechanism for driving the output pixels to the limits of the gray scale. In the language of Section 4.5.2, the non-linear constraints make interior, face, and edge fixed points unstable without affecting the stability status of corner fixed points. Orientation-sensitive halftoning performed on two other test images is illustrated in Figure 5.6 and Figure 5.7.

All three of Figure 5.5, Figure 5.6, and Figure 5.7 exhibit more of the line and curve features found in hand-drawn halftones (such as the Wall Street Journal portraits). On the CM-2, the simulation takes 3000 iterations at the time step of 0.01 sec. This amounts to about a minute of real time. We emphasize that in contrast, the actual Wall Street Journal portraits are drawn by hand, without the aid of computer, and the entire process takes an artist three to five hours [74].

5.4.2 Noise-Shaping Least-Squares Halftoning With Clipped M-Lattice System

In this section we combine the noise shaping theory and the M-lattice system concepts in order to derive a novel halftoning algorithm based on the least-squares optimization. We show that the binary images produced by the new method offer the



(a) (b)

Figure 5.6: Orientation-sensitive halftoning. (a) the original "Reagan" image; (b) the "Reagan" image adaptively halftoned using orientation information at each pixel of the original.

advantages of both the error diffusion and the least-squares halftoning algorithms.

It is generally agreed that error diffusion produces the best results in terms of artifacts. The detail rendition capabilities of error diffusion are excellent. Highlight, mid-tone, and shadow detail are all capable of being reproduced [75]. Due to the in-loop non-linearity of the one-bit comparator, the conventional error diffusion system is restricted to be causal in order to be recursively computable. Since the neighborhood of the error filter is causal, the decisions at any point in the image depend on the "past" decisions only. This causality of the error diffusion algorithm prevents it from making sharp transitions and tracking edges properly. It produces correlated artifacts of directional hysteresis that result from the raster order of processing. The serial nature of the algorithm means that the errors get propagated in the direction the image is scanned, which results in these artifacts [76], [73].

(a) (b)

Figure 5.7: Orientation-sensitive halftoning. (a) the original "Alex holding koala" image; (b) the "Alex holding koala" image adaptively halftoned using orientation information at each pixel of the original.

Different error propagation schemes have been proposed to cope with this. But this is an impossible challenge. There is no proper way to directly extend error diffusion to 2-D, because there is no natural order in 2-D. Non-causality is what error diffusion algorithms really miss [73], [77], [78]. A non-causal algorithm makes the decisions at any point in the image, depending on the "past" as well as on the "future" decisions, which can alleviate these shortcomings [71].

Generally, least-squares halftoning techniques render edges well, but suffer from granular artifacts. We show that the M-lattice system naturally combines noise shaping with least-squares optimization, thereby offering the benefits of both.

Appendix D reviews the noise shaping theory, which is the underlying principle of operation of the error diffusion algorithm and the oversampling A / D converter (used in digital audio systems) [79], [80], [81], [82], [83], [73], [84], [85]. Least-squares halftoning principles are summarized in Appendix E. Here, we use the least-squares intensity-approximation halftoning method of Section E.2 in order to illustrate that the clipped M-lattice system, (4.48), is inherently suited for incorporating the noise shaping structures into the existing least-squares halftoning techniques.

Letting $\vec{y} = \vec{\chi}$ and inserting (E.9) into (4.48) at equilibrium gives:

$$\begin{aligned}
\Phi(\vec{\chi}) &= (\mathbf{H}^T\vec{s})^T\vec{\chi} - \frac{1}{2}\vec{\chi}^T\mathbf{H}^T\mathbf{H}\vec{\chi}, && (5.13)\\
\vec{\nabla}_{\vec{\chi}}\Phi(\vec{\chi}) &= \mathbf{H}^T\vec{s} - \mathbf{H}^T\mathbf{H}\vec{\chi}, && (5.14)\\
-\mathbf{A}\vec{\psi} &= \mathbf{H}^T\vec{s} - \mathbf{H}^T\mathbf{H}\vec{\chi}, \text{ or} && (5.15)\\
-\left(\mathbf{A} - \mathbf{H}^T\mathbf{H}\right)\vec{\psi} &= \mathbf{H}^T\vec{s} - \mathbf{H}^T\mathbf{H}(\vec{\chi} - \vec{\psi}). && (5.16)
\end{aligned}$$

Now set $-\left(\mathbf{A} - \mathbf{H}^T\mathbf{H}\right) = \mathbf{I}$, and let $\vec{q} \stackrel{\text{def}}{=} \vec{\chi} - \vec{\psi}$ be the quantization error (or the quantization noise). Then (5.16) becomes:

$$\vec{\psi} = \mathbf{H}^T\vec{s} - \mathbf{H}^T\mathbf{H}\vec{q}. \qquad (5.17)$$

Thus, according to (5.17), the clipped M-lattice system performs non-causal error diffusion in the steady-state limit (see Section D.1 of Appendix D). For perceptual reasons, it is desirable to minimize the low-frequency content of the quantization error. Since $\mathbf{H}^T\mathbf{H}$ is a smoothing filter, $\mathbf{H}_Q \stackrel{\text{def}}{=} \mathbf{I} - \mathbf{H}^T\mathbf{H}$ becomes a high-pass filter. Then it follows that $\mathbf{A} = -\mathbf{H}_Q$. The action of the high-pass noise shaping filter, $\mathbf{H}_Q$, gives the quantization noise the perceptually pleasant "blue" character [69]. We exploit the fact that $\mathbf{A}$ can have off-diagonal elements by making it act as a perceptually-based filter. Therefore, the resulting images correspond to local minima that are visually more pleasant than those produced using a diagonal $\mathbf{A}$ matrix.

Starting with the equation for error diffusion, (5.17), and reversing the above steps leads to (5.14), the equation for the clipped M-lattice system in steady state. Error diffusion has been modeled as a Hopfield network that uses $q(\psi_i)$ in place of $g(\psi_i)$ [73]. However, the non-monotonicity of $q(\psi_i)$ causes instability. In contrast, slightly perturbing $\mathbf{A}$ so as to make it negative-definite guarantees that (4.48) will be stable for binary outputs. Hence, the clipped M-lattice system is a more suitable model than the Hopfield network for non-causal error diffusion.

The performance of the new M-lattice halftoning method was evaluated in comparison to the error diffusion algorithm, using the test image, shown in Figure 5.8(a). Figure 5.8(b) shows this image, halftoned by error diffusion. For the sake of simplicity and fair comparison, the low-pass filter, programmed into the clipped M-lattice system, is the symmetric non-causal version of the original Floyd & Steinberg error diffusion filter [80]. Both filters are given in Figure 5.9 [2]. The result of halftoning the original image using the clipped M-lattice system appears in Figure 5.8(c). In addition, the two close-up versions are provided in Figure 5.10 for comparison. The new method provides accurate detail rendition without introducing correlated texture.

Several key aspects of the least-squares halftoning with the clipped M-lattice system are unique. Unlike the analog Hopfield network or the cellular neural network, the clipped M-lattice system allows the $\mathbf{A}$ matrix to have off-diagonal elements. In conjunction with a quadratic quality metric, this enables the clipped M-lattice system to perform blue noise filtering as well as make the resulting halftone images optimal in the least-squares sense.

In addition, with this example we have shown that the noise shaping architectures can incorporate all the enhancements presently attributed to the flexibility

[2]This choice of a non-causal filter has no justification with regard to optimality; we believe it could be improved if effort was invested.

(a)

(b)

(c)

Figure 5.8: Faithful-rendition halftoning. (a) the original "Lena" image; (b) the "Lena" image halftoned using the original Floyd & Steinberg error diffusion algorithm; (c) the "Lena" image halftoned using the clipped M-lattice system with the symmetric non-causal version of the original Floyd & Steinberg error diffusion filter.

$$
\begin{array}{ccc}
0 & 0 & 0 \\
0 & * & 7 \\
3 & 5 & 1
\end{array}
\qquad\qquad
\begin{array}{ccc}
1 & 5 & 3 \\
7 & * & 7 \\
3 & 5 & 1
\end{array}
$$

(a) (b)

Figure 5.9: Noise-shaping filters. (a) the original Floyd & Steinberg error diffusion filter $(\times\frac{1}{16})$; (b) the symmetric non-causal version of the original Floyd & Steinberg error diffusion filter $(\times\frac{1}{32})$.

(a) (b)

Figure 5.10: Faithful-rendition halftoning. Magnification is $\times 2$ on a side. (a) the "Lena" image halftoned using the original Floyd & Steinberg error diffusion algorithm; (b) the "Lena" image halftoned using the clipped M-lattice system with the symmetric non-causal version of the original Floyd & Steinberg error diffusion filter.

of the least-squares halftoning methods. The most important strength of the latest least-squares halftoning approaches is that they can employ explicit models of the human visual system and of the printing device [86], [72].

5.5 Chapter Summary

In this chapter, we have described how the M-lattice can be applied to signal synthesis and processing tasks. The model has many modes of operation, and every application explores one or more of these different regimes.

First, we performed a "sanity-check" on the M-lattice system by confirming that it could synthesize visual textures, which look virtually indistinguishable from those generated by discretizing the original Turing's reaction-diffusion system and bounding the variables from within the numerical integration method. Then we used it to synthesize reaction-diffusion sound textures, the first of their kind. They are generated digitally with the M-lattice, but contain pleasant "analog" timbres.

Second, we identified the problem of fingerprint restoration and halftoning to be a natural application of the clipped M-lattice system, because of its ability to synthesize zebra stripes, which appear qualitatively similar to ridges and bifurcations comprising human fingerprints. Using orientation detection for designing the system parameters so as to emphasize the significant features of a fingerprint image causes the clipped M-lattice system to act as an infinitely-aggressive band-pass filter. As a result, ridges and bifurcations are extracted at the highest contrast, even if they are only faintly detectable in the original image, while scratches, unevennesses in illumination, and other extraneous detail are removed. The contrast of the restored features is enhanced with the help of the binarization capability of the clipped M-lattice system.

Third, we have applied the M-lattice to digital halftoning of images. As a non-linear programming technique, the M-lattice system is capable of solving constrained optimization problems with flexible objective functions. Orientation-sensitive halftoning makes use of this property. When the objective function is a quadratic form, the M-lattice can be designed to perform blue-noise filtering. This implies that the resulting halftone images can be made not only optimal in the least-squares sense, but also perceptually pleasant.

Chapter 6

Conclusions

If this thesis had to be summarized by one picture and one sentence, then that picture would be Figure 6.1, and that sentence would be the caption, accompanying the figure. In a nutshell, this work presents the novel "M-lattice system", developed for solving practical signal-processing problems with reaction-diffusion. It is remarkable that the same mathematical model that can synthesize animal coat markings can also generate high-quality printouts, seek local optima of non-linear objective functions, and perform a handful of other engineering applications. The next section summarizes theoretical and practical contributions of this research to the field of signal processing.

6.1 Contributions

The main theoretical contribution is the formulation of the M-lattice system, which addresses the issues of flexibility and boundedness of reaction-diffusion systems. The experimental contributions of this research make use of the theoretical results. Various modes of the M-lattice system have been demonstrated to be useful for signal processing. These contributions are stated more precisely below.

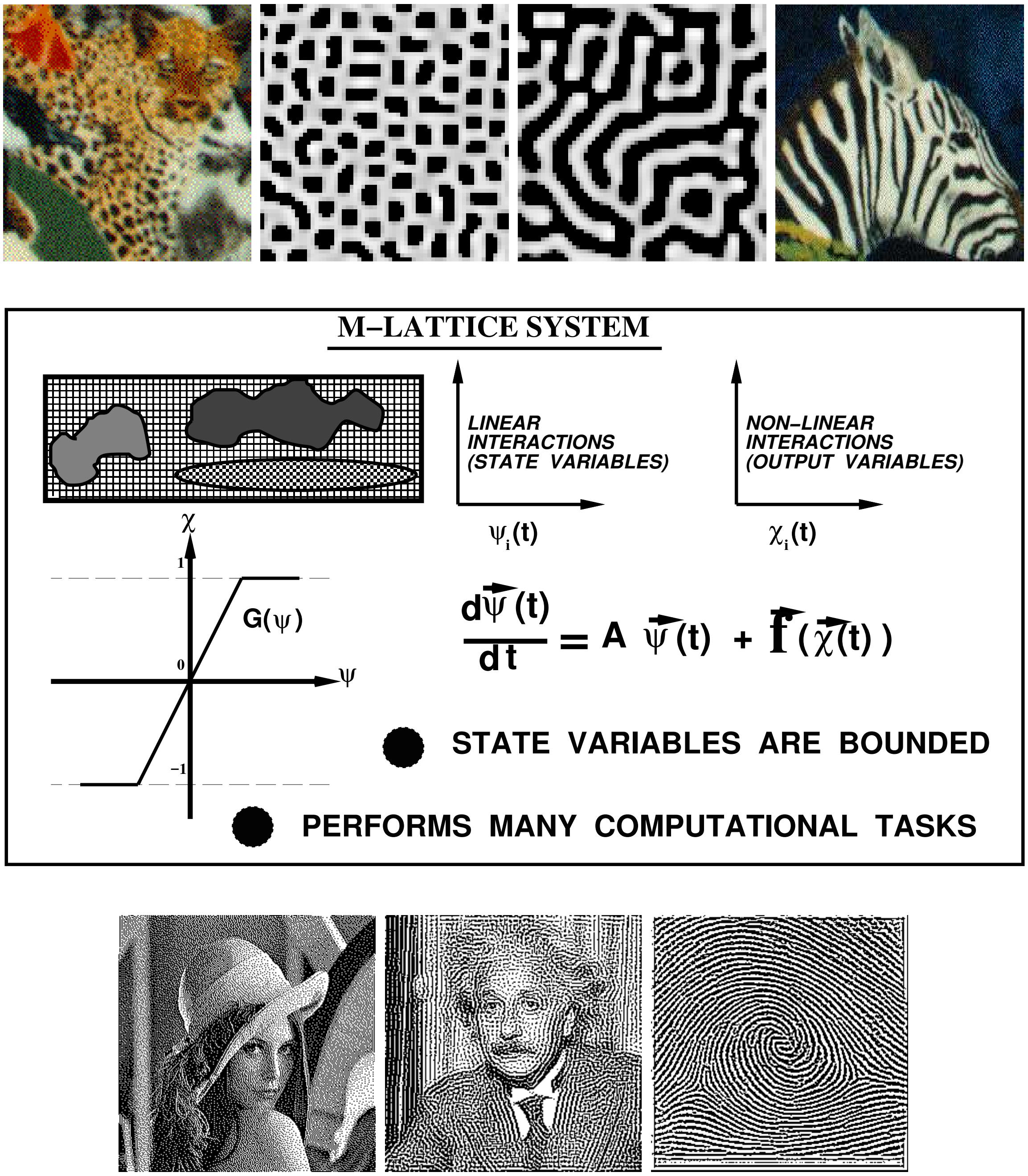


Figure 6.1: M-lattice: From spots and stripes on animals to signal processing.

6.1.1 Theoretical Contributions

Flexibility: The reaction-diffusion system restricts the interactions of its state variables (or morphogens). Specifically, the linear interactions are restricted to intra-layer diffusion. In addition, the non-linear interactions are restricted to inter-layer reaction and only among the state variables at the same position in the lattice. In contrast, the M-lattice system allows the linear and the non-linear interactions in the lattice to be both intra-layer and inter-layer. This flexibility enables the M-lattice to capture the behavior of many physical systems. (See Section 2.1 and Section 4.2.3.)

Pattern-forming property: The ability to form patterns is what makes reaction-diffusion a useful model for signal processing. By defining what it means for a non-linear dynamical system to possess the pattern-forming property, we show that the conventional reaction-diffusion systems are not the only ones that can synthesize textures. For example, based on this definition, a one-morphogen M-lattice system (*i.e.*, a 1-lattice) possesses pattern-extraction capabilities, because it can be designed to perform infinitely-sharp bandpass filtering. (See Section 2.5.)

Boundedness in practice: Reaction-diffusion lacks a mechanism for keeping its state variables bounded, which is a problem in practical applications. In contrast, the convergence of the M-lattice has been observed in computer simulation for a large variety of non-linear reaction functions. We show that under certain condition the state variables of the M-lattice system are bounded; therefore, even when the M-lattice does not converge to an equilibrium, it has "nice" numerical properties. Hence, it can be applied to practical tasks. (See Section 4.3.)

Total stability: In order to account for some of these observations, we prove the total stability of a subclass of the M-lattice system. This subclass is called the diagonal-state clipped M-lattice system and is a generalization of the cellular neural network. While the cellular neural network is totally stable, the proof, given by its inventors, contains imprecise statements and omissions. The proof of the total stability of the diagonal-state clipped M-lattice system applies also to the cellular neural network. (See Section 4.5.1.)

Relationships to established models: The proposed M-lattice system is closely related to the analog Hopfield network and the cellular neural network, but has more flexibility in how its variables interact. (See Section 4.4, Section 4.4.3, and Section 4.5.1.)

Binary outputs: An important problem is how to cause the outputs of various types of the M-lattice to assume binary values in equilibrium. We develop the sufficient conditions on the parameter values and study the limiting case of an infinite sigmoidal gain. The analysis concerns the stability of fixed points, making precise an earlier argument based on circuit diagrams that was published for the case of the cellular neural network. For the special case of the analog Hopfield network, the analysis here explains why the output variables converge to binary values if the neuron auto-connections are non-negative. In other words, in order to cause binary outputs, the neuron auto-connections do not necessarily have to be strictly zero, but can also be positive. Observing this behavior in computer simulation has puzzled researchers for a period of time, and, to the best of our knowledge, this result provides the first theoretical account. (See Section 4.5.2.)

Non-linear optimization: We have classified the kinds of non-linear optimization that various subclasses of the M-lattice can perform and how they relate to the optimization aspects of the established models, such as the analog Hopfield network and the non-linear programming neural network. For binary-valued outputs, the M-lattice, like the analog Hopfield network, is capable of optimization in the sense of the Hamming distance of one for objective functions that are multilinear polynomials. For real-valued outputs, the clipped M-lattice performs optimization in the standard sense and provides a realistic mathematical model for the non-linear programming neural network circuit. (See Section 4.4.3 and Section 4.6.)

Non-causal-neighborhood error diffusion: In addition to simulating reaction-diffusion, the M-lattice possesses other signal-processing aspects. For example, we have shown that the M-lattice is the most appropriate dynamical model for the non-causal-neighborhood error diffusion halftoning algorithm. (See Section 5.4.2.)

Blue-noise halftones as local optima: This modeling provides a parity condition that allows one to check whether or not the equilibrium binary image is a local minimum in the sense of the Hamming distance of one of a certain quadratic cost functional. The fact that error diffusion performs blue-noise filtering together with optimization is a possible explanation of the high quality of the resulting halftones. (See Section 5.4.2.)

6.1.2 Practical Contributions

Signal synthesis: First, we verified that the M-lattice can simulate basic reaction-diffusion by duplicating a number of published visual textures. In addition, we have used the M-lattice to synthesize sound textures, samples of which have been played by the author in a computer music performance. (See Section 2.6 and Section 5.1.)

“Special-effects” halftoning: The fact that the M-lattice can perform constrained optimization is explored in a new halftoning algorithm. By incorporating orientation information into a quality metric and relying on explicit constraints to enforce binary values, we have synthesized pictures of Einstein, Reagan, and others in the creatively hand-drawn halftoning style of the Wall Street Journal portraits. To the best of our knowledge, this is the first method for doing this type of halftoning automatically (as opposed to doing it by hand). (See Section 5.4.1.)

“Faithful-rendition” halftoning: Using the fact that the M-lattice is a natural choice for modeling the non-causal-neighborhood error diffusion halftoning algorithm, we have applied the M-lattice to the problem of synthesizing binary images that are perceptually close to the original images. Thus, in addition to optimizing a quadratic objective function, this system performs blue-noise filtering. The resulting halftones exhibit excellent detail rendition without the annoying correlated artifacts found in the halftones generated by causal-neighborhood error diffusion. (See Section 5.4.2.)

Restoration and halftoning of fingerprints: The pattern-forming property makes reaction-diffusion a natural model for the preservation of ridges and bifurcations in fingerprint images. We have applied the pattern-forming property of the one-morphogen M-lattice to the restoration of the significant detail in fingerprints. The contrast of the restored features is enhanced with the help of binarization capability of the M-lattice. (See Section 5.3.)

Chapter 7

Future Work

Throughout this thesis report, we have marveled at the rich diversity of operating regimes demonstrated by reaction-diffusion and the M-lattice. Chapter 5 describes the details of how a subset of these modes of operation is explored in practical signal-processing applications. In the present chapter, we propose two new applications of the M-lattice that were not researched in depth as part of this thesis. The first application is texture restoration and employs the real-valued non-linear programming regime of the M-lattice. The second application is data encryption and uses the chaotic regime of the M-lattice.

7.1 Texture Restoration

Consider an image restoration problem, where the image in question contains several textured regions, possibly corrupted by noise. In this problem, the goal is to recover each of the uncorrupted texture regions. If the regions are supported by different models, then this leads to a difficult non-linear problem.

Figure 7.1 illustrates the proposed feature-based approach for texture enhancement. The feature vector is judiciously chosen so as to be able to sort the given collection of textured images into a sufficiently diverse set of classes. The elements of the feature vector can have different scales. The only requirement is that each feature is represented by a real number. The textured image to be enhanced is divided into regions, sufficiently large for computing the features reliably, but sufficiently small for the enhancement to adapt to the given sampling resolution. The classification / enhancement process is performed at every region concurrently.

Each dashed box encloses the model of one texture class. The distance metric determines how close the features of the observed degraded image are to those of the textures that would hypothetically be generated by all the models. The Gaussian switch produces ≈ 1 if the two compared textures have similar features and ≈ 0 if their features differ significantly. Assuming that at most one of the models agrees with the observation, the winning model's low-level texture synthesis algorithm will be selected to enhance the degraded image.

The specific mechanism by which the models communicate with each other and trade off the amount of synthetic texture for the actual texture observed in the degraded image is based on the regularization principle. For each model's low-level texture synthesis algorithm, there is an associated penalty term for not using that algorithm. Selecting a particular model implies a certain amount of prior knowledge about the underlying texture, and neglecting that knowledge in favor of being faithful to the observed data carries a penalty.

It is important to emphasize that the entire optimization process goes on in feature space, so there is no need for synthesizing the enhanced texture explicitly, until the end. At the very end, when the equilibrium is reached, the enhanced synthetic image is generated from the features computed for each region.

While the proposed scheme has a "winner-take-all" flavor, the Gaussian switch

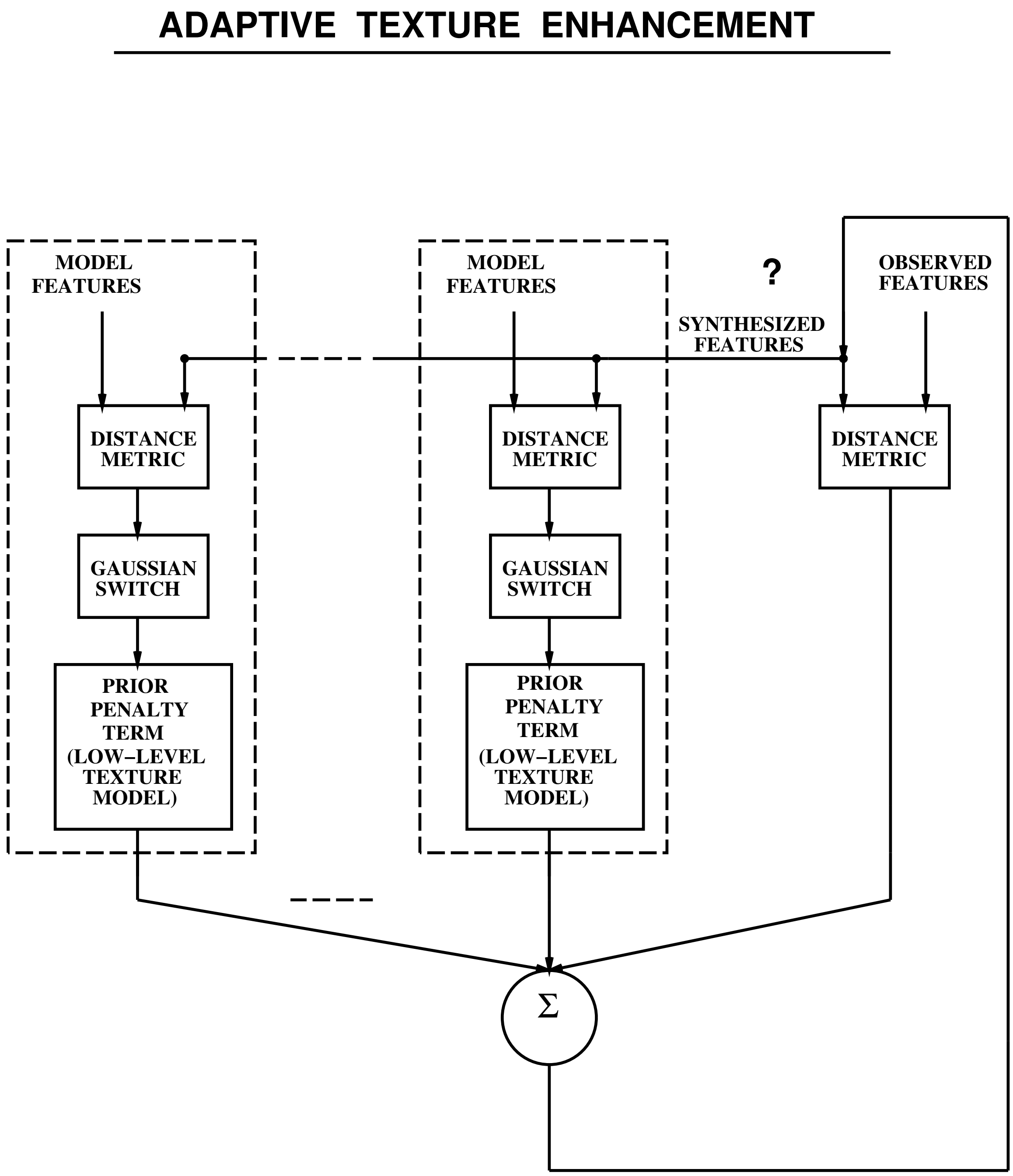


Figure 7.1: Texture classification / enhancement using the M-lattice system.

will not produce binary outputs for all regions, given a real-world enhancement task. Thus, if the texture synthesis algorithms can be carefully tuned to depend on the features of the neighboring regions and not only compete but also cooperate with one another, then it should be possible to generate synthetic images that do not suffer from artificial abrupt texture boundaries. Since the feature space is continuous and since the Gaussian switches are not binary threshold elements, the features of the observed image will mix with the features produced by the cooperating and competing texture synthesis algorithms [87]. We believe that while facilitating the dominance of the texture prevalent in a given region, this smoothness will also provide for a natural blending together of neighboring textured regions in the processed image.

In Section 4.6.2, it was shown that if the objective function of a continuous vector-valued variable is bounded, then the M-lattice system can be used for optimizing this objective function. It remains to make an argument for choosing distance metrics and penalty terms such that the total cost functional is bounded and has a minimum; otherwise, the problem has no solution. There is no no benefit in allowing distance metrics or penalty terms to produce greater discrepancies than a certain upper bound, chosen in a way that renders the corresponding textures virtually unrelated. If no class can be found to model the texture in the given region accurately, then there are two possible outcomes. One choice is to leave the synthesized image as an approximation of the observed degraded image, because all of the Gaussian switches will be open. Since the Gaussian function rolls off rapidly, there is no sense in making its argument significantly greater than is necessary to turn the switch off. The other choice is to signal a "class unknown" error message.

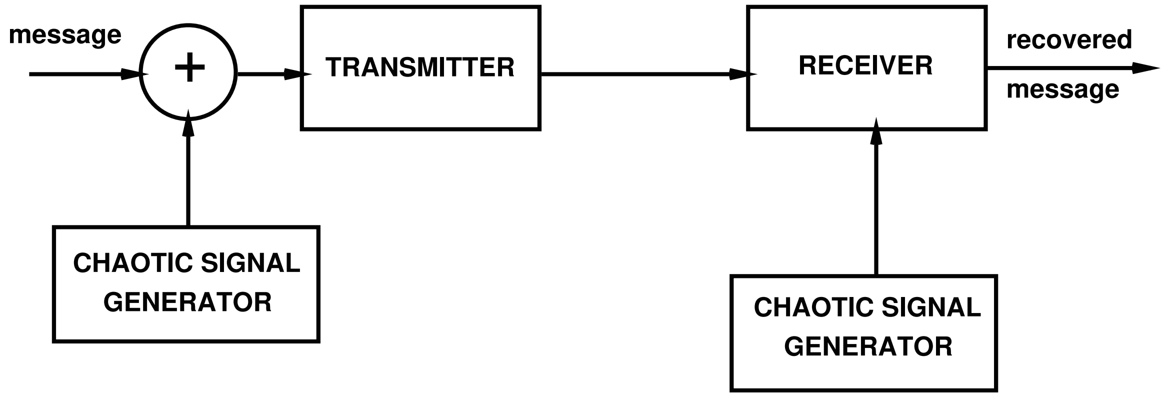


Figure 7.2: A data encryption / decryption scheme that employs a chaotic system.

7.2 Communication Using Chaos In M-Lattice

Normally, one thinks of chaos as a fascinating curiosity at best, and a nuisance at worst, something to be avoided or engineered away [88], [22]. But people have found ways to exploit chaos to do some practical things. For example, recent discoveries in the theory of chaotic systems have led researchers to novel methods for communicating secret messages [89], [90], [91]. The encryption / decryption scheme is illustrated in Figure 7.2. According to the diagram, the useful message signal is added to a high-amplitude waveform produced by a chaotic system. This scrambled signal is then transmitted. The mixture is unscrambled on the receiving end with the help of an identical chaotic system. The two chaotic systems are synchronized by connecting the second chaotic signal to an appropriate circuit node in the receiver [89]. The original message is then recovered by subtracting the synchronized chaotic waveform from the scrambled signal [90], [91] [1].

[1] Many people would doubt the marvel that two chaotic systems could be synchronized. After all, chaotic systems are sensitive to slight changes in the initial conditions, so one might expect any errors between the transmitter and the receiver to grow exponentially [89], [22].

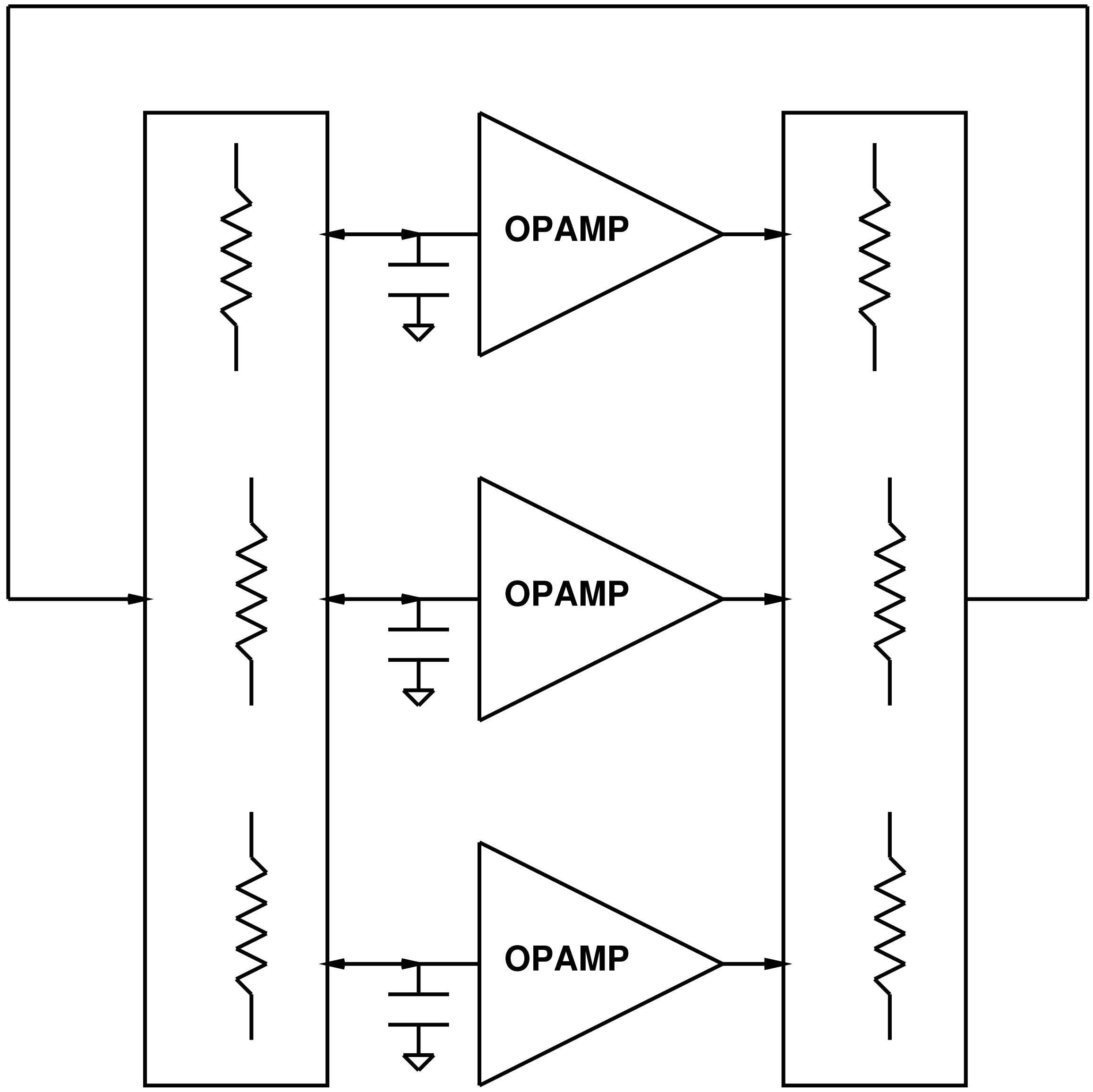


Figure 7.3: Chaotic 3-lattice circuit. Operational amplifiers are the only required non-linear elements.

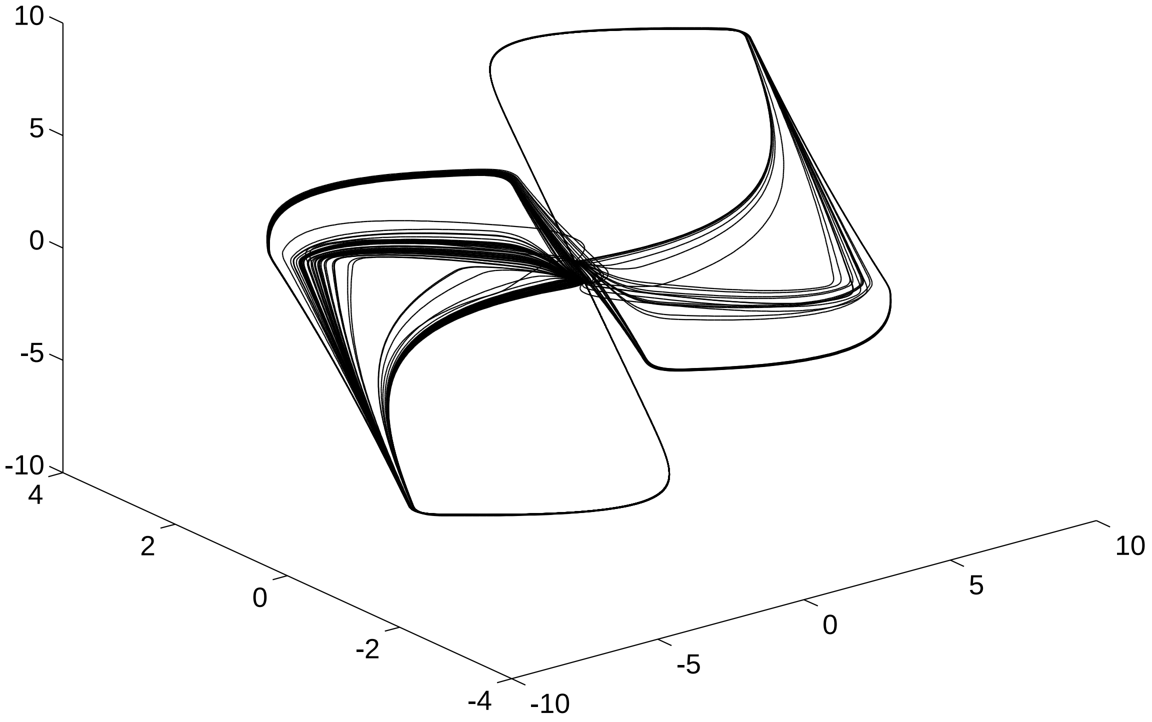


Figure 7.4: Chaos in the M-lattice system. The plot shows the "strange" attractor in the state-space.

The first actual circuit built on this principle [90] uses the chaotic system, described in the pioneering work by Lorenz [92]. Lorenz's system contains the quadratic non-linearity, whose implementation requires analog multipliers.

Here, we propose a chaotic circuit based on the M-lattice system that uses only operational amplifiers and linear resistors (see Figure 7.3). Each operational amplifier plays the role of the warping function, and this is the only source of non-linearity needed in the circuit. Preliminary computer experiments indicate that the following clipped 3-lattice exhibits chaos:

$$\frac{d\vec{\psi}(t)}{dt} = \mathbf{A}\vec{\psi}(t) + \mathbf{B}\vec{\chi}(t), \tag{7.1}$$

$$\mathbf{A} = \frac{1}{4}\begin{bmatrix} -9 & 4 & 4 \\ 4 & -9 & 4 \\ 4 & 4 & -9 \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} 9 & -3 & -6 \\ -5 & 6 & -1 \\ -1 & -9 & 10 \end{bmatrix}, \quad T = \frac{1}{4}. \tag{7.2}$$

The "strange" attractor of this clipped M-lattice system, (7.1), is shown in Figure 7.4. The points were plotted only after the transients have decayed.

7.3 Chapter Summary

In this chapter, we have suggested two additional applications of the M-lattice for future research. One is the adaptive texture restoration; it relies on the ability of the M-lattice to perform the real-valued non-linear programming. The other is data encryption; it explores the chaotic regime of the M-lattice.

Bibliography

[1] A. M. Turing, "The chemical basis of morphogenesis," *Philosophical Transactions of Royal Society of London*, vol. 237, no. B, pp. 37–72, 1952.

[2] "Private communications with Lenore Ramm, Internet and Cambridge, MA," Oct. 1992.

[3] S. A. Kauffman, *The Origins Of Order: Self-Organization And Selection In Evolution.* New York, NY: Oxford University Press, 1993.

[4] J. B. L. Bard and I. Lauder, "How well does Turing's theory of morphogenesis work?," *Journal of Theoretical Biology*, vol. 45, pp. 501–531, 1974.

[5] Y. Xu, C. M. Vest, and J. D. Murray, "Holographic interferometry used to demonstrate a theory of pattern formation in animal coats," *Applied Optics*, vol. 22, pp. 3479–3483, 1983.

[6] H. Meinhardt, *Models Of Biological Pattern Formation.* New York, NY: Academic Press, 1982.

[7] D. A. Young, "A local activator-inhibitor model of vertebrate skin patterns," *Mathematical Biosciences*, vol. 72, pp. 51–58, 1984.

[8] J. D. Murray, "A pre-pattern formation mechanism for animal coat markings," *Journal of Theoretical Biology*, vol. 88, pp. 161–199, 1981.

[9] J. D. Murray, "On pattern formation mechanisms for lepidopteran wing patterns and mammalian coat markings," *Philosophical Transactions of Royal Society of London*, vol. 295, no. B, pp. 473–496, 1981.

[10] J. D. Murray, "Parameter space for Turing instability in reaction diffusion mechanisms: A comparison of models," *Journal of Theoretical Biology*, vol. 98, pp. 143–163, 1982.

[11] N. F. Britton, *Reaction-Diffusion Equations and Their Applications to Biology.* New York, NY: Academic Press, 1986.

[12] J. D. Murray, *Mathematical Biology.* Berlin: Springer-Verlag, 1993.

[13] J. D. Murray, "How the leopard gets its spots," *Scientific American*, vol. 258, pp. 80–87, Mar. 1988.

[14] A. Witkin and M. Kass, "Reaction-diffusion textures," *Computer Graphics*, vol. 25, pp. 299–308, July 1991.

[15] G. Turk, "Generating textures on arbitrary surfaces using reaction-diffusion," *Computer Graphics*, vol. 25, pp. 289–298, July 1991.

[16] C. B. Price, P. Wambacq, and A. Oosterlinck, "Applications of reaction-diffusion equations to image processing," in *Third International Conference on Image Processing and its Applications*, (Warwick, UK), July 1989.

[17] "Private communications with David Rees, CSIRO, Australia," May 1992.

[18] "Private communications with Christian Bruteanu, CSIRO, Australia," May 1992.

[19] P. Perona and J. Malik, "Scale-space and edge detection using anisotropic diffusion," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 12, pp. 629–639, July 1990.

[20] M. Nitzberg and T. Shiota, "Nonlinear image filtering with edge and corner enhancement," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 14, pp. 826–833, Aug. 1992.

[21] L. A. Segel, *Mathematical Models In Molecular And Cellular Biology.* Cambridge, UK: Cambridge University Press, 1980.

[22] S. H. Strogatz, *Nonlinear Dynamics And Chaos: With Applications To Physics, Biology, Chemistry, And Engineering.* Reading, MA: Addison-Wesley, 1994.

[23] G. Strang, *Introduction to Applied Mathematics.* Cambridge, MA: Wellesley-Cambridge Press, 1986.

[24] J. B. L. Bard, "A model for generating aspects of zebra and other mammalian coat patterns," *Journal of Theoretical Biology*, vol. 93, pp. 363–385, 1981.

[25] J. S. Lim, *Two-Dimensional Signal and Image Processing.* New Jersey: Prentice-Hall, 1990.

[26] A. V. Oppenheim and R. W. Schafer, *Digital Signal Processing.* London: Prentice-Hall, Inc., 1975.

[27] "Private communications with George C. Verghese, MIT," Dec. 1993.

[28] "Private communications with Jean-Jacques E. Slotine, MIT," May 1994.

[29] C. B. Price, P. Wambacq, and A. Oosterlinck, "Image enhancement and analysis with reaction-diffusion paradigm," *IEE Proceedings*, vol. 137, June 1990.

[30] J. E. Marsden and A. J. Tromba, *Vector Calculus.* New York: W. H. Freeman and Company, 1981.

[31] G. Strang, *Linear Algebra and Its Applications.* Academic Press, 1980.

[32] W. Rudin, *Principles of Mathematical Analysis.* New York: McGraw-Hill, Inc., 1976.

[33] M. W. Hirsch, "Convergence in neural nets," in *Proceedings Of IEEE International Joint Conference on Neural Networks*, pp. II:115–125, 1987.

[34] F. M. A. Salam, Y. Wang, and M.-R. Choi, "On the analysis of dynamic feedback neural nets," *IEEE Transactions on Circuits and Systems*, vol. 38, pp. 196–201, Feb. 1991.

[35] M. Vidyasagar, "Minimum-seeking properties of analog neural networks with multilinear objective functions," Tech. Rep. (Unpublished), Centre For AI & Robotics, Bangalore, India, 1993.

[36] J. J. Hopfield, "Neural networks and physical systems with emergent collective computational abilities," *Proceedings of National Academy of Sciences, USA*, vol. 79, pp. 2554–2558, Apr. 1982.

[37] J. J. Hopfield, "Neurons with graded response have collective computational properties like those of two-state neurons," *Proceedings of National Academy of Sciences, USA*, vol. 81, pp. 3088–3092, May 1984.

[38] J. J. Hopfield and D. W. Tank, ""Neural" computation of decisions in optimization problems," *Biological Cybernetics*, vol. 52, pp. 141–152, 1985.

[39] L. O. Chua and G.-N. Lin, "Nonlinear programming without computation," *IEEE Transactions on Circuits and Systems*, vol. 31, pp. 182–188, Feb. 1984.

[40] M. P. Kennedy and L. O. Chua, "Unifying the Tank and Hopfield linear programming circuit and the canonical nonlinear programming circuit of chua and lin," *IEEE Transactions on Circuits and Systems*, vol. 34, pp. 210–214, Feb. 1987.

[41] M. P. Kennedy and L. O. Chua, "Neural networks for nonlinear programming," *IEEE Transactions on Circuits and Systems*, vol. 35, pp. 554–562, May 1988.

[42] L. O. Chua and L. Yang, "Cellular neural networks: Theory," *IEEE Transactions on Circuits and Systems*, vol. 35, pp. 1257–1272, Oct. 1988.

[43] L. O. Chua and L. Yang, "Cellular neural networks: Applications," *IEEE Transactions on Circuits and Systems*, vol. 35, pp. 1273–1290, Oct. 1988.

[44] T. Roska and L. O. Chua, "Cellular neural networks with nonlinear and delay-type template elements," in *Proc. IEEE Int. Workshop on Cellular Neural Networks and Applications*, (Budapest, Hungary), 1990.

[45] L. O. Chua and T. Roska, "Stability of a class of nonreciprocal cellular neural networks," *IEEE Transactions on Circuits and Systems*, vol. 37, pp. 1520–1527, Dec. 1990.

[46] L. O. Chua, T. Roska, *et al.*, "Various publications on cellular neural networks," *IEEE Transactions on Circuits and Systems*, 1988–1993.

[47] W. E. Boyce and R. C. DiPrima, *Elementary Differential Equations And Boundary Value Problems.* New York: Wiley, 1977.

[48] M. Vidyasagar, *Nonlinear Systems Analysis.* New Jersey: Prentice-Hall, 1978.

[49] J. H. Wilkinson, *The Algebraic Eigenvalue Problem.* Oxford: Clarendon Press, 1965.

[50] G. H. Golub and C. F. V. Loan, *Matrix Computations.* Baltimore, Maryland: The Johns Hopkins University Press, 1983.

[51] C. Mead, *Analog VLSI and Neural Systems.* Addison-Wesley, 1989.

[52] B. W. Lee and B. J. Sheu, *Hardware Annealing in Analog VLSI Neurocomputing.* Kluwer Academic Publishers, 1991.

[53] M. S. Bazaraa and C. M. Shetty, *Nonlinear Programming.* New York: John Wiley and Sons, 1979.

[54] "Private communications with Tod Machover, MIT Media Laboratory," Dec. 1992.

[55] H. D. Hubel and T. N. Wiesel, "Receptive fields and functional architecture of monkey striate cortex," *J. Physiology*, vol. 195, pp. 215–243, 1968.

[56] W. T. Freeman and E. H. Adelson, "The design and use of steerable filters," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-13, pp. 891–906, Sept. 1991.

[57] R. W. Picard and M. Gorkani, "Finding perceptually dominant orientations in natural textures," *Spatial Vision, Spec. Julesz Issue*, 1993. To Appear; also avail. as Percep. Comp. TR #229, M.I.T. Media Lab.

[58] A. P. Russo, "An automatic system for fingerprint analysis," Master's thesis, Rensselaer Polytechnic Institute, Troy, NY, 1986.

[59] A. Sherstinsky and R. W. Picard, "Restoration and enhancement of fingerprint images using M-Lattice – a novel non-linear dynamical system," in *Proceedings of International Conference on Pattern Recognition (To Appear)*, Oct. 1994.

[60] "Private communications with Aaron Bobick, MIT," May 1994.

[61] A. Papoulis, *Probability, Random Variables, and Stochastic Processes*. McGraw-Hill, 1965.

[62] R. O. Duda and P. E. Hart, *Pattern Classification and Scene Analysis*. New York: Wiley, 1973.

[63] S. Geman and D. Geman, "Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-6, no. 6, pp. 721–741, 1984.

[64] A. K. Katsaggelos, *Digital Image Restoration*. Berlin, Germany: Springer Verlag, 1991.

[65] R. W. Picard, *Texture Modeling: Temperature Effects on Markov/Gibbs Random Fields*. ScD thesis, M.I.T., 1991.

[66] "Private communications with Dmitry B. Utyansky, Novintech-Leintec, Ltd.," Feb. 1994.

[67] M. Kass and A. Witkin, "Analyzing oriented patterns," in *Readings In Comp. Vision* (M. A. Fischler and O. Firschein, eds.), Morgan Kaufman, 1987.

[68] G. Zorpette, "A horse of a different color," *IEEE Spectrum*, pp. 17–18, July 1992.

[69] R. A. Ulichney, *Digital Halftoning*. Cambridge, MA: MIT Press, 1987.

[70] A. Sherstinsky and R. W. Picard, "M-Lattice: A novel non-linear dynamical system and its application to halftoning," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (To Appear)*, Apr. 1994.

[71] A. Sherstinsky, "Feasibility of model-based non-causal-neighborhood halftoning," *Written portion of the Area Examination, EECS Graduate Area III, MIT*, Dec. 1992.

[72] T. N. Pappas and D. L. Neuhoff, "Least-squares model-based halftoning," in *Proc. SPIE Conf. on Electronic Imaging Science and Technology*, (San Jose, CA), Feb. 1992.

[73] D. Anastassiou, "Error diffusion coding for A/D conversion," *IEEE Transactions on Circuits and Systems*, vol. 36, pp. 1175–1186, Sept. 1989.

[74] "Private communications with Richard Kelman, Dow Jones," May 1994.

[75] J. C. Stoffel and J. F. Moreland, "A survey of electronic techniques for pictorial image reproduction," *IEEE Transactions on Communications*, vol. COM-29, pp. 1898–1925, Dec. 1981.

[76] D. Anastassiou, "Neural net based digital halftoning of images," in *Proceedings 1988 International Symposium on Circuits and Systems*, (Helsinki, Finland), June 1988.

[77] T. Bernard, "From Σ-Δ modulation to digital halftoning of images," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*, 1991.

[78] J. B. Mulligan and J. Albert J. Ahumada, "Principled halftoning based on human vision models," *SPIE Vol. 1666 Human Vision, Visual Processing, and Digital Display*, 1992.

[79] H. Inose, Y. Yasuda, and J. Murakami, "A telemetering system by code modulation – Δ-Σ modulation," *IRE Trans. Space Electronics and Telemetry*, vol. SET-8, pp. 204–209, Sept. 1962.

[80] R. W. Floyd and L. Steinberg, "An adaptive algorithm for spatial grey scale," in *SID Int. Sym. Dig. Tech. Papers*, pp. 36–37, 1975.

[81] J. C. Candy, "The structure of quantization noise from sigma-delta modulation," *IEEE Transactions on Communications*, vol. COM-29, pp. 1316–1323, Sept. 1981.

[82] J. C. Candy, "Decimation for sigma delta modulation," *IEEE Transactions on Communications*, vol. COM-34, pp. 72–76, Jan. 1986.

[83] R. M. Gray, "Oversampled sigma-delta modulation," *IEEE Transactions on Communications*, vol. COM-35, pp. 481–489, May 1987.

[84] K. H.Chao, S. Nadeem, W. L. Lee, and C. G. Sodini, "A Higher Order Topology for Interpolative Modulators for Oversampling A/D Converters," *IEEE Transactions on Circuits and Systems*, vol. 37, pp. 309–318, Mar. 1990.

[85] A. Sherstinsky and C. G. Sodini, "A Programmable Demodulator for Oversampled Analog-to-Digital Modulators," *IEEE Transactions on Circuits and Systems*, vol. 37, pp. 1092–1103, Sept. 1990.

[86] T. N. Pappas and D. L. Neuhoff, "Model-based halftoning," in *Proc. SPIE Conf. on Electronic Imaging Science and Technology*, (San Jose, CA), Feb. 1991.

[87] A. Sherstinsky and R. W. Picard, "On the efficiency of the orthogonal least squares training method for radial basis function networks," Tech. Rep. 271, M.I.T. Media Lab Perceptual Computing Group, Oct. 1993. To Appear in IEEE Transactions on Neural Networks.

[88] J. Gleick, *Chaos: Making A New Science.* New York, NY: Viking, 1987.

[89] L. M. Pecora and T. L. Carrol, "Synchronization in chaotic systems," *Physical Review Letters*, vol. 64, no. 821, 1990.

[90] K. M. Cuomo and A. V. Oppenheim, "Cicruit implementation of synchronized chaos, with applications to communications," *Physical Review Letters*, vol. 71, no. 65, 1993.

[91] K. M. Cuomo, A. V. Oppenheim, and S. H. Strogatz, "Synchronization of lorenz-based chaotic cicruits, with applications to communications," *IEEE Transactions on Circuits and Systems*, vol. 40, pp. 626–633, Oct. 1993.

[92] E. N. Lorenz, "Deterministic nonperiodic flow," *Journal of Atmospheric Science*, vol. 20, no. 130, 1963.

[93] R. A. Ulichney, "Dithering with blue noise," *Proceedings of the IEEE*, vol. 76, pp. 56–79, Jan. 1988.

[94] A. Zakhor, S. Liu, and F. Eskafi, "A new class of B/W and color halftoning algorithms," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*, 1991.

[95] P. Carnevali, L. Coletti, and S. Patarnello, "Image processing by simulated annealing," in *Proc. in Computer Vision* (M. A. Fischler and O. Firschein, eds.), Morgan Kaufmann, 1987.

Appendix A

Glossary

A.1 Mathematical Symbols

We will state the notation for the case of continuous space, $\vec{x}$, and continuous time, t. For the case of discrete space, $\vec{x}$ is replaced by (*e.g.*, in 2-D) $\vec{n} = (n_x, n_y)$, and for the case of discrete time, t is replaced by n_t. For every discrete variable that does not have a continuous counterpart, the notation will be stated explicitly.

$\Re$**:** Set of real numbers.

$\Re_+$**:** Set of positive real numbers.

$\mathcal{D}$**:** General derivative operator.

$\vec{x}$**:** Position vector in continuous space.

$\vec{n} = (n_x, n_y)$**:** Position vector in discrete (2-D) space.

t**:** Continuous time.

n_t: Discrete time.

m: Morphogen index.

M: Total number of morphogens.

d: Number of spatial dimensions.

$\psi_m(\vec{x}, t)$: Concentration of the m-th morphogen as a function of space and time.

$\vec{\psi}(\vec{x}, t)$: Vector of M morphogen concentrations, $\psi_m(\vec{x}, t)$, $m = 1, \ldots, M$.

$R_m(\vec{\psi}(\vec{x}, t))$: Reaction function of the m-th morphogen, which can depend on all M morphogen concentrations at $\vec{x}$ and t.

$\mathbf{R}(\vec{n})$: Coupling matrix due to reaction for morphogens in a linear-reaction reaction-diffusion system or a convolutionally-coupled M-lattice system as a function of discrete space.

$\mathbf{B}(\vec{n})$: Coupling matrix due to diffusion for morphogens in a linear-reaction reaction-diffusion system or a convolutionally-coupled M-lattice system as a function of discrete space.

$\mathbf{A}(n_x, n_y)$: Overall coupling matrix for morphogens in a linear-reaction reaction-diffusion system or a convolutionally-coupled M-lattice system as a function of discrete space.

$\vec{k} = (k_x, k_y)$: Discrete 2-D spatial frequency, or spatial-frequency index, or spatial wave number, *etc.*

$\vec{\Psi}(\vec{k}, t)$: Vector of M morphogen concentrations as a function of discrete spatial frequency and continuous time.

$\mathbf{R}(\vec{k})$: Coupling matrix due to reaction for morphogens in a linear-reaction reaction-diffusion system or a convolutionally-coupled M-lattice system as a function of discrete spatial frequency.

$\mathbf{B}(\vec{k})$: Coupling matrix due to diffusion for morphogens in a linear-reaction reaction-diffusion system or a convolutionally-coupled M-lattice system as a function of discrete spatial frequency.

$\mathbf{A}(\vec{k})$: Overall coupling matrix for morphogens in a linear-reaction reaction-diffusion system or a convolutionally-coupled M-lattice system as a function of discrete spatial frequency.

$\vec{H}(k_x, k_y, t)$: Transfer function of a linear-reaction reaction-diffusion system or a convolutionally-coupled M-lattice system as a function of discrete 2-D spatial frequency and continuous time.

When reaction-diffusion systems are generalized to M-lattice systems, we change the notation, used for morphogen concentrations, as follows:

$\psi_i(t)$: State variable at discrete index, i, and a function of continuous time, t.

$\vec{\psi}(t) = [\psi_1(t),\ \ldots,\ \psi_N(t)]^T$: Vector of N state variables, $\psi_i(t)$.

$G(\cdot)$: Warping function (such as, for example, a sigmoid), used to constrain or bound the state variables.

T: Reciprocal of the slope of $G(\cdot)$, also known as the "temperature".

$\chi_i(t)$: Output variable, obtained by warping $\psi_i(t)$ with $G(\cdot)$, at discrete index, i, and a function of continuous time, t.

$\vec{\chi}(t) = [\chi_1(t),\ \ldots,\ \chi_N(t)]^T$: Vector of N output variables, $\chi_i(t)$.

$\psi(\vec{n}, t)$: State variable at a discrete spatial position, $\vec{n}$, and a function of continuous time, t.

$\chi(\vec{n}, t)$: Output variable, obtained by warping $\psi(\vec{n}, t)$ with $G(\cdot)$, at a discrete spatial position, $\vec{n}$, and a function of continuous time, t.

$\psi(\vec{k}, t)$: State variable at a discrete spatial-frequency position, $\vec{k}$, and a function of continuous time, t.

$\chi(\vec{k}, t)$: Output variable, obtained by warping $\psi(\vec{n}, t)$ with $G(\cdot)$, at a discrete spatial-frequency position, $\vec{k}$, and a function of continuous time, t.

$\vec{a}(\vec{\psi}(t))$: Functional specification for the interactions among the state variables.

$\vec{f}(\vec{\chi}(t))$: Functional specification for the interactions among the output variables.

$\vec{\nabla}_{\vec{\psi}}$: Gradient with respect to the elements of $\vec{\psi}$.

$\vec{\nabla}_{\vec{\chi}}$: Gradient with respect to the elements of $\vec{\chi}$.

$\Phi(\vec{\chi}(t))$: Objective function.

$\vec{\nabla}_{\vec{\chi}}\Phi(\vec{\chi}(t))$: Common functional form of the interactions among the output variables.

$\mathbf{J}_{\vec{c}}(\vec{\psi})$: Jacobian of $\vec{c}(\vec{\psi}(t))$ at $\vec{\psi}$.

A.2 Terminology

morphogenesis: Formation of distinctive animal coat patterns, colorings, and markings. Also, formation of shape in low life forms. (See Chapter 1.)

reaction-diffusion system: Any reaction-diffusion system with possibly non-linear reaction. (See Definition 2.1.1 and Definition 2.5.1.)

linear-reaction reaction-diffusion system: This is a reaction-diffusion system with reaction restricted to be linear. (See Definition 3.1.1.)

M-lattice system: Any system that supports linear interactions among its state variables and non-linear interactions among its output variables. The state and the output variables can be organized arbitrarily. The system's name is inspired by its roots in reaction-diffusion, where M is the number of morphogens, or layers in the lattice. Indeed, any and all sites of the M-layered lattice, or the M-lattice for short, can be arbitrarily organized into M layers. (See Definition 4.2.1.)

clipped M-lattice system: This system is an M-lattice system, in which the warping function is the clipping non-linearity. (See Definition 4.5.1.)

linear M-lattice system: This system is an M-lattice system with interactions restricted to be linear. (See Chapter 2.)

convolutionally-coupled M-lattice system: This is a linear M-lattice system with interactions restricted to convolutions with linear filters. (See Definition 3.2.1.)

evocator: Waveform of small random perturbations. (See Chapter 2.)

Appendix B

Mathematical Background

In order to make this document self-contained, we have summarized some key mathematical preliminaries.

B.1 Dynamical Systems And Stability

Most of the material in this section has been adapted from standard texts on the subject [47], [48].

A general vector differential equation has the form:

$$\frac{d\vec{\psi}(t)}{dt} = \vec{f}(t, \vec{\psi}(t)), \; t > 0, \tag{B.1}$$

where $\vec{\psi}(t) \in \Re^N$ and $\vec{f}: \; \Re_+ \times \Re^N \longrightarrow \Re^N$. The present discussion is restricted to non-linear autonomous systems, *i.e.*, systems described by:

$$\frac{d\vec{\psi}(t)}{dt} = \vec{f}(\vec{\psi}(t)). \tag{B.2}$$

This system is called autonomous, because the right hand side of (B.2) does not explicitly depend on the time variable, t.

Definition B.1.1 *A fixed (or critical, or equilibrium,* etc.*) point of the autonomous system, (B.2), is any point,* $\vec{\psi}$*, which is a solution of:*

$$\vec{f}(\vec{\psi}) \;=\; \vec{0}. \tag{B.3}$$

We assume throughout that (B.2) has at least one fixed point, $\vec{\psi}$.

Definition B.1.2 *The fixed point,* $\vec{\psi}$*, of (B.2) is said to be stable if, for each* $\varepsilon > 0$*, there exists a* $\delta(\varepsilon) > 0$ *such that:*

$$\left\|\vec{\psi}(t_0) - \vec{\psi}\right\| < \delta(\varepsilon) \;\Longrightarrow\; \left\|\vec{\psi}(t) - \vec{\psi}\right\| < \varepsilon,\ \forall\ t \geq t_0. \tag{B.4}$$

Definition B.1.3 *The fixed point,* $\vec{\psi}$*, is unstable if it is not stable.*

Note that the term "instability" does not mean that the trajectory of the system has to "blow up" in a sense that $\left\|\vec{\psi}(t)\right\| \to \infty$ as $t \to \infty$. While this is one possibility, it is not the only one. Instability simply requires that for some $\varepsilon > 0$, no $\delta > 0$ can be found such that (B.4) holds [48].

Definition B.1.4 *The fixed point,* $\vec{\psi}$*, of (B.2) is asymptotically stable if:*

1. *It is stable; and*

2. *For any* ε*, there exists a* $\delta > 0$*, such that:*

$$\left\|\vec{\psi}(t_0) - \vec{\psi}\right\| < \delta \;\Longrightarrow\; \left\|\vec{\psi}(t) - \vec{\psi}\right\| < \varepsilon,\ \forall\ t > \tau(\varepsilon) > t_0. \tag{B.5}$$

Both parts of Definition B.1.4 are essential. In particular, the first condition is crucial, because the second condition does not imply stability [48].

Associated with every asymptotically stable fixed point is its "region of attraction", defined as follows:

$$\mathcal{S}(\vec{\psi}) \;=\; \left\{\vec{\psi}(t_0) \in \Re^N : \left\|\vec{\psi}(t) - \vec{\psi}\right\| < \varepsilon,\ t > \tau(\varepsilon) > t_0\right\}. \tag{B.6}$$

All trajectories, starting at time t_0 from an initial state within $\mathcal{S}$, eventually converge to $\vec{\psi}$.

Definition B.1.5 *The fixed point, $\vec{\psi}$, of (B.2) is globally asymptotically stable if:*

$$\left\| \vec{\psi}(t) - \vec{\psi} \right\| < \varepsilon, \text{ whenever } t > \tau(\varepsilon) > t_0, \tag{B.7}$$

regardless of what $\vec{\psi}(t_0)$ is.

The global asymptotic stability results if all trajectories of the system converge to the fixed point, $\vec{\psi}$, as $t \to \infty$. In other words, the region of attraction of a globally asymptotically stable fixed point is the entire space, $\Re^N$ [48].

Stability, asymptotic stability, and instability are local concepts, dealing with the trajectories of the system in the vicinity of a fixed point, whereas global asymptotic stability, as the name implies, is a global concept, having to do with the behavior of all the trajectories of the system.

Thus, global asymptotic stability is a property of the system as a whole. On the contrary, the notions of stability, instability, and asymptotic stability pertain to a specific fixed point of the system, because in general a system can have more than one fixed point, each of which has its own set of stability properties [48].

Requiring global asymptotic stability, the strongest kind of stability, is often too restrictive and unnecessary. In fact, many practical dynamical systems are not globally asymptotically stable, but have many asymptotically stable fixed points. Since one system of the latter kind is the focus of the present research, we now define one last stability category in order to be able to make a global-type statement about an entire system in terms of some of its local fixed points, simultaneously.

Definition B.1.6 *A trivial trajectory corresponds to $\vec{\psi}(t_0) = \vec{\psi}$.*

Definition B.1.7 *The autonomous system, (B.2), is called totally stable if its every non-trivial trajectory approaches a finite asymptotically stable fixed point, $\vec{\psi}$, for any finite initial conditions, $\vec{\psi}(t_0)$.*

Note that Definition B.1.7 does not imply that every fixed point of (B.2) has to be asymptotically stable, or even stable. The only requirement is that every solution trajectory must approach an equilibrium.

Totally stable dynamical systems are interesting for engineering purposes. Not only are they guaranteed to converge to some fixed point, but they also provide mechanisms for guiding the trajectory of the system to the desired fixed point. Typically, this is accomplished by controlling the parameters of the system and specifying the initial conditions.

B.1.1 Auxiliary Functions

Here, we review positive-definite and other related functions. Various literature sources discuss these concepts, some on a more rigorous level than others [47], [48]. The goal of the following alternative treatment is to provide intuition for analyzing the stability of, specifically, the M-lattice type of systems, using the Lyapunov theory.

Definition B.1.8 *Throughout this definition, let:*

$$\left.\begin{array}{rcl} \vec{x} & \in & \Re^N; \\ \vec{p} & \in & \mathcal{S} \subset \Re^N; \\ c & \in & \Re; \\ & \mathcal{S} & \textit{is an open set.} \end{array}\right\} \tag{B.8}$$

A continuous function $V : \Re^N \longrightarrow \Re$ *on* $\mathcal{S}$ *is said to be a:*

- *locally positive-semidefinite function if:*

$$\left\{\begin{array}{rcl} V(\vec{p}) & = & c, \\ V(\vec{x}) & \geq & c, \ \forall\ \vec{x} \neq \vec{p},\ \|\vec{x}\| \in \mathcal{S}; \end{array}\right. \tag{B.9}$$

- *locally positive-definite function if:*

$$\begin{cases} V(\vec{p}) &= c, \\ V(\vec{x}) &> c, \ \forall \ \vec{x} \neq \vec{p}, \ \|\vec{x}\| \in \mathcal{S}; \end{cases} \tag{B.10}$$

- *globally positive-semidefinite function if:*

$$\begin{cases} V(\vec{p}) &= c, \\ V(\vec{x}) &\geq c, \ \forall \ \vec{x} \neq \vec{p}, \\ V(\vec{x}) &\rightarrow \infty, \ as \ \|\vec{x}\| \rightarrow \infty; \end{cases} \tag{B.11}$$

- *globally positive-definite function if:*

$$\begin{cases} V(\vec{p}) &= c, \\ V(\vec{x}) &> c, \ \forall \ \vec{x} \neq \vec{p}, \\ V(\vec{x}) &\rightarrow \infty, \ as \ \|\vec{x}\| \rightarrow \infty; \end{cases} \tag{B.12}$$

- *locally negative-semidefinite function if:*

$$\begin{cases} V(\vec{p}) &= c, \\ V(\vec{x}) &\leq c, \ \forall \ \vec{x} \neq \vec{p}, \ \|\vec{x}\| \in \mathcal{S}; \end{cases} \tag{B.13}$$

- *locally negative-definite function if:*

$$\begin{cases} V(\vec{p}) &= c, \\ V(\vec{x}) &< c, \ \forall \ \vec{x} \neq \vec{p}, \ \|\vec{x}\| \in \mathcal{S}; \end{cases} \tag{B.14}$$

- *globally negative-semidefinite function if:*

$$\begin{cases} V(\vec{p}) &= c, \\ V(\vec{x}) &\leq c, \ \forall \ \vec{x} \neq \vec{p}, \\ V(\vec{x}) &\rightarrow -\infty, \ as \ \|\vec{x}\| \rightarrow \infty; \end{cases} \tag{B.15}$$

- *globally negative-definite function if:*

$$\begin{cases} V(\vec{p}) &= c, \\ V(\vec{x}) &< c, \ \forall \ \vec{x} \neq \vec{p}, \\ V(\vec{x}) &\rightarrow -\infty, \ as \ \|\vec{x}\| \rightarrow \infty. \end{cases} \tag{B.16}$$

B.1.2 Lyapunov Functions

The classical definition of a Lyapunov function is in terms of locally positive-definite functions and locally negative-semidefinite functions, where the local domain is $\mathcal{S} = \mathcal{S}(\vec{\psi})$ near a fixed point, $\vec{\psi}$.

Definition B.1.9 *The function $V(\vec{x}(t))$ is called a Lyapunov function if $V(\vec{x}(t))$ is continuous with continuous first partial derivatives (C^1), is a locally positive-definite function, and $\dfrac{dV(\vec{x}(t))}{dt}$ is a locally negative-semidefinite function with $c = 0$ [47], [48].*

Of course, if $V(\vec{x}(t))$ is C^1, is a locally negative-definite function, and $\dfrac{dV(\vec{x}(t))}{dt}$ is a locally positive-semidefinite function with $c = 0$, then such a $V(\vec{x}(t))$ is also a Lyapunov function.

There are no separate definitions for cases, in which local semidefiniteness is strengthened to local definiteness. But, naturally, locally-definite functions that fit Definition B.1.9 are also regarded as Lyapunov functions. Numerous examples of Lyapunov functions abound [47], [48], [22].

B.1.3 Lyapunov's Direct Method

For notational convenience, let $\vec{\psi} = \vec{p} = \vec{0}$ and $c = 0$. There is no loss of generality, since any function will acquire a point at the origin upon a suitable change of coordinates and / or translation. The following theorem and proof are adapted from [47].

Theorem B.1.1 *Suppose that the autonomous system, (B.2), has an isolated fixed point at the origin. If a Lyapunov function, $V(\vec{\psi}(t))$, defined with respect to the origin, can be associated with the system, then the origin is a stable fixed point. If, in addition, $\dfrac{dV(\vec{\psi}(t))}{dt}$ is a locally negative-definite function with $c = 0$, then the origin is an asymptotically stable fixed point.*

Proof: Let $\dfrac{dV(\vec{\psi}(t))}{dt}$ be a locally negative-semidefinite function. Let $k \geq 0$ be a constant and consider the curve in the $x1 - x2$ plane given by $V(\vec{\psi}(t)) = k$. For $k = 0$, the curve reduces to the single point, $\vec{\psi} = \vec{0}$. However, for $k > 0$ and sufficiently small, it can be shown by using the continuity of $V(\vec{\psi}(t))$ that the curve is a closed curve, containing the origin, as illustrated in Figure B.1(a). We assume further that if $0 < k_1 < k_2$, then the curve $V(\vec{\psi}(t)) = k_1$ lies within the curve $V(\vec{\psi}(t)) = k_2$. We show that a trajectory starting inside a closed curve $V(\vec{\psi}(t)) = k$ cannot cross to the outside. Thus, given a circle of radius ε about the origin, by taking k sufficiently small we can ensure that every trajectory starting inside the closed curve $V(\vec{\psi}(t)) = k$ stays within the circle of radius ε; indeed, it stays within the closed curve $V(\vec{\psi}(t)) = k$ itself. Thus, the origin is a stable fixed point.

To show this, note that since $V(\vec{\psi}(t)) = k$ defines a level set of $V(\vec{\psi}(t))$, $\vec{\nabla}_{\vec{\psi}} V(\vec{\psi}(t))$ is normal to the curve $V(\vec{\psi}(t)) = k$ and points in the direction of increasing $V(\vec{\psi}(t))$. In the present case, $V(\vec{\psi}(t))$ increases outward from the origin, so $\vec{\nabla}_{\vec{\psi}} V(\vec{\psi}(t))$ points away from the origin, as indicated in Figure B.1(b). Next, consider the trajectory, $\vec{\psi}(t)$ of the system, (B.2), and note that the vector, $\dfrac{d\vec{\psi}(t)}{dt}$ is tangent to the trajectory at each point. Let $t = \tau$ be the time at which $\vec{\psi}(t)$ intersects a closed curve $V(\vec{\psi}(t)) = k$. At this time, we have for (B.2):

$$\left.\frac{d\vec{\psi}(t)}{dt}\right|_{t=\tau} = \left.\vec{f}(\vec{\psi}(t))\right|_{t=\tau}. \tag{B.17}$$

So differentiating $V(\vec{\psi}(t))$ with respect to time, t, gives:

$$\left.\frac{dV(\vec{\psi}(t))}{dt}\right|_{t\,=\,\tau} \;=\; \left[\vec{\nabla}_{\vec{\psi}} V(\vec{\psi}(t))\right]\Big|_{t\,=\,\tau}^{T} \left.\left[\frac{d\vec{\psi}(t)}{dt}\right]\right|_{t\,=\,\tau}. \tag{B.18}$$

Since $\left.\frac{dV(\vec{\psi}(t))}{dt}\right|_{t\,=\,\tau} \leq 0$, it follows that the cosine of the angle between $\left[\vec{\nabla}_{\vec{\psi}} V(\vec{\psi}(t))\right]\Big|_{t\,=\,\tau}$ and $\left.\left[\frac{d\vec{\psi}(t)}{dt}\right]\right|_{t\,=\,\tau}$ is also less than or equal to zero; hence, the angle itself is in the range $\left[\frac{\pi}{2}, \frac{3\pi}{2}\right]$. Thus, the direction of motion on the trajectory is inward with respect to $V(\vec{\psi}(t)) = k$ or, at worst, tangent to this curve. Trajectories starting inside a closed curve $V(\vec{\psi}(t)) = k$ (no matter how small k is) cannot escape, so the origin is a stable fixed point. If $\left.\frac{dV(\vec{\psi}(t))}{dt}\right|_{t\,=\,\tau} < 0$, then the trajectories passing through points on the curve are actually pointed inward. As a consequence, it can be shown that trajectories starting sufficiently close to the origin must approach the origin; hence, the origin is an asymptotically stable fixed point [47]. ∎

In order to extend Theorem B.1.1 to global asymptotic stability, a Lyapunov function, $V(\vec{\psi}(t))$, must be a globally positive-definite function, and its time derivative must be a locally negative-definite function with $\mathcal{S} = \Re^N$. The global definiteness of $V(\vec{\psi}(t))$ is a key necessary condition for global asymptotic stability: it is absolutely essential that $V(\vec{x}) \to \infty$ as $\|\vec{x}\| \to \infty$. Otherwise, if $V(\vec{\psi}(t))$ is only locally-definite, then it is still possible for $V(\vec{\psi}(t))$ to be a Lyapunov function on $\mathcal{S}$, but fail to be a Lyapunov function outside of $\mathcal{S}$, thereby allowing $\|\vec{\psi}(t)\| \to \infty$ as $t \to \infty$. On the contrary, this problem cannot occur if there exists a globally-definite Lyapunov function, $V(\vec{\psi}(t))$, such that $\frac{dV(\vec{\psi}(t))}{dt}$ is a locally negative-definite function with $\mathcal{S} = \Re^N$. By Theorem B.1.1, the very fact that such a Lyapunov function is associated with the dynamical system, (B.2), means that there is no mechanism for $\|\vec{\psi}(t)\|$ to grow unbounded.

Note that for global asymptotic stability, $V(\vec{\psi}(t))$ is required to be globally-

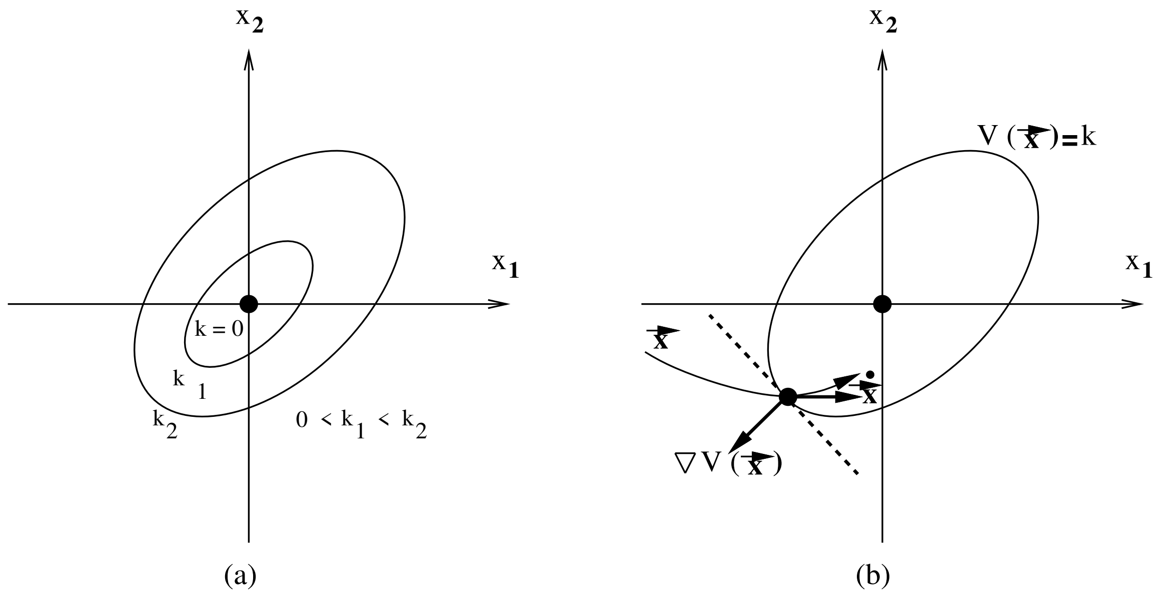


Figure B.1: Geometrical interpretation in 2-D of Lyapunov's direct method.

definite, but $\dfrac{dV(\vec{\psi}(t))}{dt}$ is not required to be globally-definite. For total stability, neither $V(\vec{\psi}(t))$ nor $\dfrac{dV(\vec{\psi}(t))}{dt}$ is required to be globally-definite. However, it is still necessary to have $\|\vec{\psi}(t)\| < \infty$.

B.1.4 Gradient Systems

Finally, we discuss the issue of determining the total stability of a system. A well-known difficulty with using Lyapunov's direct method for analyzing dynamical systems is that there are no general guidelines for coming up with Lyapunov functions. Yet, in order to determine whether or not a given dynamical system is totally stable in the sense of Lyapunov, a valid Lyapunov function must be conjured up for every fixed point of interest. However, in most practical dynamical systems, the number of stable fixed points is rather large, and having to treat each fixed point separately is undesirable. Fortunately, certain kinds of totally stable systems possess convenient built-in collective features that allow the proof of total stability to be constructed using a single function that has the Lyapunov properties near the stable equilibria. "Gradient" systems, discussed next, possess such characteristics [34].

Definition B.1.10 *The system, (B.2), is called a gradient system if there exists a* C^1 *function,* $V(\vec{\psi}(t))$*, with the property:*

$$-\vec{\nabla}_{\vec{\psi}} V(\vec{\psi}(t)) = \frac{d\vec{\psi}(t)}{dt} = \vec{f}(\vec{\psi}(t)). \tag{B.19}$$

Many gradient systems originated in physics. Thus, the function, $V(\vec{\psi}(t))$, is frequently called the "potential" or the "energy", and the negative of its gradient, $-\vec{\nabla}_{\vec{\psi}} V(\vec{\psi}(t))$, is commonly referred to as the "field".

Lemma B.1.1 *Gradient systems do not contain cycles.*

Proof: We compute the time derivative of $V(t) \stackrel{\text{def}}{=} V(\vec{\psi}(t))$ by the chain rule:

$$\begin{aligned}
\frac{dV(\vec{\psi}(t))}{dt} &= -\left[\vec{\nabla}_{\vec{\psi}} V(\vec{\psi}(t))\right]^T \left[\frac{d\vec{\psi}(t)}{dt}\right] \\
&= -\left[\frac{d\vec{\psi}(t)}{dt}\right]^T \left[\frac{d\vec{\psi}(t)}{dt}\right] = -\left\|\frac{d\vec{\psi}(t)}{dt}\right\|^2 < 0.
\end{aligned} \tag{B.20}$$

In order for a cycle to exist, the integral of $\frac{dV(\vec{\psi}(t))}{dt}$ over the cycle must vanish. However, (B.20) shows that this is impossible. ∎

According to Definition B.1.10, the class of gradient systems is highly restrictive. Nevertheless, the following relaxed form of this definition applies to a number of practical dynamical systems, such as the analog Hopfield network and the reciprocal cellular neural network.

Definition B.1.11 *The system, (B.2), is called an almost-gradient system if there exists a* C^1 *function,* $V(\vec{\psi}(t))$*, with the property:*

$$\vec{\nabla}_{\vec{\psi}} V(\vec{\psi}(t)) = \vec{0} \iff \frac{d\vec{\psi}(t)}{dt} = \vec{f}(\vec{\psi}(t)) = \vec{0}. \tag{B.21}$$

According to Definition B.1.11, the fixed points of an almost-gradient system must be the extrema of $V(\vec{\psi}(t))$. But the right hand side of (B.2) does not have to be the gradient of $V(\vec{\psi}(t))$.

Thus, if it can be shown that, regardless of initial conditions, all solution trajectories are eventually confined to a closed ball in the $\vec{\psi}$-space, then providing a decreasing and bounded-below (or an increasing and bounded-above) auxiliary function becomes sufficient in order to ascertain the system's total stability.

The following statement is a consequence of Definition B.1.8, Definition B.1.9, and Theorem B.1.1, but it does not seem to be formulated anywhere in this form.

Corollary B.1.1 *Consider an almost-gradient autonomous system, (B.2), together with an auxiliary function,* $V(\vec{\psi}(t))$*, associated with the system. Let an open ball,* $\mathcal{B}_r$*, of radius* r *be defined as follows:*

$$\mathcal{B}_r \stackrel{\text{def}}{=} \left\{ \vec{\xi} \in \Re^N,\ r \in \Re_+ :\ \|\vec{\xi}\| < r \right\} \tag{B.22}$$

and suppose that the following conditions are satisfied:

$$\left. \begin{array}{c} V(\vec{\psi}(t)) \in \mathrm{C}^1, \\ \forall\ t > t_0 :\ \dfrac{dV(\vec{\psi}(t))}{dt} < 0, \\ \forall\ \vec{x} \in \Re^N\ \exists\ \beta \in \Re :\ V(\vec{x}) \geq \beta; \\ \forall\ \vec{\psi}(t_0)\ \exists\ r = R \in \Re_+ :\ \lim\limits_{t \to \infty} \vec{\psi}(t) \in \overline{\mathcal{B}}_R, \end{array} \right\} \tag{B.23}$$

where $\overline{\mathcal{B}}_R$ *is the closure of* $\mathcal{B}_R$*. Then this almost-gradient system is totally stable.*

Proof: Definition B.1.11 and the assumptions in (B.23) imply that there can be three types of fixed points:

1. The fixed point, $\vec{\psi}$, is a local maximum of $V(\vec{\psi}(t))$. In this case, $V(\vec{\psi}(t))$ is a locally negative-definite function. Hence, $\vec{\psi}$ is unstable [47], [48].

2. The fixed point, $\vec{\psi}$, is a saddle of $V(\vec{\psi}(t))$. In this case, $V(\vec{\psi}(t))$ is not a Lyapunov function for $\vec{\psi}$, and no conclusion regarding the stability of $\vec{\psi}$ can be made.

3. The fixed point, $\vec{\psi}$, is a local minimum of $V(\vec{\psi}(t))$. In this case, $V(\vec{\psi}(t))$ is a Lyapunov function for $\vec{\psi}$. Hence, $\vec{\psi}$ is asymptotically stable by Theorem B.1.1.

The conditions in (B.23) guarantee that $V(\vec{\psi}(t))$ has at least one minimum. On one hand, all solution trajectories are eventually confined to a closed ball, $\overline{\mathcal{B}}_R$, in the $\vec{\psi}$-space. On the other hand, $V(\vec{\psi}(t))$ is decreasing and bounded-below. Hence, $V(\vec{\psi}(t))$ has a minimum on $\overline{\mathcal{B}}_R$.

Finally, the fact that $V(\vec{\psi}(t))$ is strictly decreasing implies that the solution trajectories do not contain cycles. Since $V(\vec{\psi}(t))$ is a continuous function of the state variables, the curves of constant $V(\vec{\psi}(t))$ (level sets) are closed curves [30]. The system's trajectory starts at the contour, corresponding to $\vec{\psi}(t_0)$ and always moves away from the starting contour, without the possibility of return.

Note that one can choose $V(\vec{\psi}(t))$ to be increasing and bounded-above by reversing the direction of the corresponding inequalities in (B.23). Then the local minima of $V(\vec{\psi}(t))$ become the unstable fixed points of the system, the local maxima of $V(\vec{\psi}(t))$ become the stable fixed points of the system, and the rest of proof remains valid.

Thus, the given almost-gradient system has at least one stable fixed point and does not exhibit cycles. Therefore, it is totally stable. ∎

B.2 Linear Algebra

The material of this section has been adapted from standard texts on the subject [49], [31], [50], [23].

B.2.1 Gerschgorin's Theorems

Certain aspects of our work with non-linear dynamical systems rely on two theorems, which are due to Gerschgorin (1931). The following theorem and proof are adapted from [49] and [31].

Theorem B.2.1 *Every eigenvalue, λ, of the matrix, $\mathbf{A}$, lies in at least one of the circular discs with centers, a_{ii}, and radii, $r_i = \sum_{j \neq i} |a_{ij}|$.*

Proof: Since λ is an eigenvalue of $\mathbf{A}$, there is certainly at least one non-zero $\vec{x}$, such that:

$$\begin{aligned} \mathbf{A}\vec{x} &= \lambda\vec{x}, \text{ or} && \text{(B.24)} \\ (\mathbf{A} - \lambda\mathbf{I})\,\vec{x} &= \vec{0}. && \text{(B.25)} \end{aligned}$$

Suppose that the i-th component of $\vec{x}$ has the largest modulus. Then expanding (B.25) at the index, i, and using the triangle inequality gives:

$$\sum_j a_{ij}x_j - \lambda x_i = 0, \tag{B.26}$$

$$a_{ii}x_i + \sum_{j\neq i} a_{ij}x_j - \lambda x_i = 0, \tag{B.27}$$

$$(\lambda - a_{ii})x_i = \sum_{j\neq i} a_{ij}x_j, \tag{B.28}$$

$$|(\lambda - a_{ii})\, x_i| = \left|\sum_{j\neq i} a_{ij}x_j\right|, \tag{B.29}$$

$$|\lambda - a_{ii}| \cdot |x_i| \leq \sum_{j\neq i} |a_{ij}x_j|\,, \tag{B.30}$$

$$|\lambda - a_{ii}| \cdot |x_i| \leq \sum_{j\neq i} |a_{ij}| \cdot |x_j|, \tag{B.31}$$

$$|\lambda - a_{ii}| \leq \sum_{j\neq i} |a_{ij}| \cdot \frac{|x_j|}{|x_i|}; \tag{B.32}$$

$$\frac{|x_j|}{|x_i|} \leq 1; \tag{B.33}$$

$$|\lambda - a_{ii}| \leq \sum_{j\neq i} |a_{ij}| = r_i. \tag{B.34}$$

Since x_i (the component that has the largest modulus) can be found for every eigenvector, $\vec{x}$, every corresponding eigenvalue, λ, lies in at least one of the circular discs [49], [31], [50], [23]. ∎

Note that Theorem B.2.1 does not guarantee the existence of a separate Gerschgorin disc for each eigenvalue. The second theorem gives more detailed information concerning the distribution of the eigenvalues among the discs.

Theorem B.2.2 *If s of the circular discs of Theorem B.2.1 form a connected domain, which is isolated from the other discs, then there are precisely s eigenvalues of* **A** *within this connected domain.*

Proof: The proof is dependent on the notion of continuity. Express **A** as follows:

$$\mathbf{A} = \mathrm{Diag}\,\{a_{11},\ \ldots,\ a_{NN}\} + \mathbf{C} = \mathbf{D} + \mathbf{C}, \tag{B.35}$$

where $\mathbf{C}$ is the matrix of off-diagonal elements. Let $r_i = \sum_{j \neq i} |a_{ij}|$ as in Theorem B.2.1 and note that $r_i = \sum_{j \neq i} |c_{ij}|$.

Consider now the range of matrices:

$$\mathbf{B}(\varepsilon) = \mathbf{D} + \varepsilon \mathbf{C}, \tag{B.36}$$

$$\text{where } 0 \leq \varepsilon \leq 1. \tag{B.37}$$

From (B.36), we have $\mathbf{B}(0) = \mathbf{D}$ and $\mathbf{B}(1) = \mathbf{A}$. The coefficients of the characteristic polynomial of $\mathbf{B}(\varepsilon)$ are polynomials in ε, and by the theory of algebraic functions, the roots of the characteristic equation are continuous functions of ε. By Theorem B.2.1, for any value of ε, the eigenvalues all lie in the circular discs with centers, a_{ii}, and radii, εr_i, and if we let ε vary steadily from 0 to 1, then the eigenvalues all traverse continuous paths.

Without loss of generality, we may assume that it is the first s discs that form the connected domain. Then, since $(N - s)$ discs with radii $r_{s+1},\ r_{s+2},\ \ldots,\ r_N$ are isolated from those with radii $r_1,\ r_2,\ \ldots,\ r_s$, the same is true for the corresponding discs with radii εr_i for all ε in the range given by (B.37). Now, when $\varepsilon = 0$, the eigenvalues are $a_{11},\ a_{22},\ \ldots,\ a_{NN}$, and of these, the first s lie in the domain corresponding to the first s discs, and the remaining $(N - s)$ lie outside this domain. It follows that this is true for all ε up to and including $\varepsilon = 1$.

In particular, if any of the Gerschgorin discs is isolated, then it contains precisely one eigenvalue. Note that corresponding results may be obtained by working with $\mathbf{A}^T$ instead of with $\mathbf{A}$ [49], [50]. ∎

B.3 Discretization Of Differential Equations

The material of this section has been adapted from standard texts on the subject, such as Strang [23].

B.3.1 $\mathcal{D}$ in 2-D

Here, we show how to discretize $\mathcal{D}$ on a 2-D spatial grid anisotropically and with an arbitrary choice of principal axes. First, we derive the continuous-space expression and then its discrete-space representation.

Continuous form

Diffusion can follow an uneven profile and an arbitrary orientation. We make use of the full Hessian matrix:

$$H = \begin{bmatrix} h_{11} & h_{12} \\ h_{21} & h_{22} \end{bmatrix} = \begin{bmatrix} \frac{\partial^2}{\partial x^2} & \frac{\partial^2}{\partial x \partial y} \\ \frac{\partial^2}{\partial y \partial x} & \frac{\partial^2}{\partial y^2} \end{bmatrix}, \tag{B.38}$$

of which the Laplacian,

$$\nabla^2 = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}, \tag{B.39}$$

is a special case. Let

$$A = \begin{bmatrix} a_x & 0 \\ 0 & a_y \end{bmatrix} \tag{B.40}$$

be the diagonal matrix of the square roots of the diffusion constants for the case where the principal directions of the process lie along the x and the y axes. For the isotropic case, $a_x = a_y = a$. Then the isotropic Laplacian diffusion is recovered from

$$\mathcal{D} \;=\; \text{trace}\left[A^T H A\right] = a^2 \nabla^2. \tag{B.41}$$

Equation (B.41) still holds for $a_x \neq a_y$, producing

$$\begin{aligned} \mathcal{D} = \text{trace}\left[A^T H A\right] &= a_x^2 \frac{\partial^2}{\partial x^2} + a_y^2 \frac{\partial^2}{\partial y^2} \\ &= a_x^2 h_{11} + a_y^2 h_{22}. \end{aligned} \tag{B.42}$$

In order to change the orientation of the principal axes, we rotate the columns of A through an angle θ using the standard rotation matrix,

$$Q \;=\; \begin{bmatrix} c & -s \\ s & c \end{bmatrix}, \tag{B.43}$$

where $c = \cos(\theta)$ and $s = \sin(\theta)$. Using QA in (B.42) together with the fact that $h_{21} = h_{12}$ yields:

$$\begin{aligned} \mathcal{D} &= \text{trace}\left[(QA)^T H (QA)\right] \\ &= a_x^2(h_{11}c^2 + 2h_{12}sc + h_{22}s^2) + a_y^2(h_{11}s^2 - 2h_{12}sc + h_{22}c^2) \\ &= h_{11}(a_x^2 c^2 + a_y^2 s^2) + h_{22}(a_x^2 s^2 + a_y^2 c^2) + 2h_{12}(a_x^2 - a_y^2)sc. \end{aligned} \tag{B.44}$$

Discrete form

Now we derive the discrete form of (B.44). We adopt the following notation:

- k is the time step;
- h is the space step [1];

[1] There should be no confusion here between the space step and the elements of the Hessian because the former is devoid of a subscript.

- Z is the temporal operator defined as follows:

$$Z\psi_m(n_x, n_y, n_t) = \psi_m(n_x, n_y, n_t + 1) \tag{B.45}$$

$$Z^{-1}\psi_m(n_x, n_y, n_t) = \psi_m(n_x, n_y, n_t - 1); \tag{B.46}$$

- K_x and K_y are the spatial operators defined as follows:

$$K_x\psi_m(n_x, n_y, n_t) = \psi_m(n_x + 1, n_y, n_t) \tag{B.47}$$

$$K_x^{-1}\psi_m(n_x, n_y, n_t) = \psi_m(n_x - 1, n_y, n_t) \tag{B.48}$$

$$K_y\psi_m(n_x, n_y, n_t) = \psi_m(n_x, n_y + 1, n_t) \tag{B.49}$$

$$K_y^{-1}\psi_m(n_x, n_y, n_t) = \psi_m(n_x, n_y - 1, n_t); \tag{B.50}$$

- I is the identity operator, which does not affect the function:

$$I\psi_m(n_x, n_y, n_t) = \psi_m(n_x, n_y, n_t). \tag{B.51}$$

Using these operators, the elements of the Hessian are discretized as follows:

$$h_{11} = \frac{K_x - 2I + K_x^{-1}}{h^2}, \tag{B.52}$$

$$h_{22} = \frac{K_y - 2I + K_y^{-1}}{h^2}, \tag{B.53}$$

$$h_{12} = \left(\frac{K_x - K_x^{-1}}{2h}\right)\left(\frac{K_y - K_y^{-1}}{2h}\right) \tag{B.54}$$

$$= \frac{K_xK_y - K_x^{-1}K_y - K_xK_y^{-1} + K_x^{-1}K_y^{-1}}{4h^2}. \tag{B.55}$$

This derivation explains the corresponding formulas stated in [14].

B.3.2 Finite Difference Methods

Using $\dfrac{Z-I}{k}$ as the discrete-time version of the time derivative, the forward Euler (FE) method for numerically solving the partial differential equation (2.2) is:

$$\psi_m(n_x, n_y, n_t+1) \quad = \quad \psi_m(n_x, n_y, n_t) + k\mathcal{D}\psi_m(n_x, n_y, n_t) + k\mathbf{R}_m(n_x, n_y, n_t), \qquad \text{(B.56)}$$

where $\mathcal{D}$ is given by (B.44). The FE method is explicit and thus simple to implement; however, it is only conditionally stable, depending on the values of k and h.

On the other hand, the backward Euler (BE) method is unconditionally stable, regardless of the values of k and h:

$$\psi_m(n_x, n_y, n_t+1) \quad = \quad \psi_m(n_x, n_y, n_t) + k\mathcal{D}\psi_m(n_x, n_y, n_t+1) + k\mathbf{R}_m(n_x, n_y, n_t+1). \qquad \text{(B.57)}$$

Since the BE method is implicit, its implementation is more involved. In fact, if a system of M morphogens is simulated on a discrete lattice $\mathcal{S}_m$ of size $|\mathcal{S}_m| = N$, then $M \times N$ non-linear equations must be solved simultaneously for each n_t [23].

B.4 Multilinear Polynomials

The material of this section has been adapted from Vidyasagar [35].

B.4.1 Basic Definitions

Definition B.4.1 *The function* $\Theta(\vec{\psi})$ *on* $\Re^N$ *is called a multilinear polynomial if* $\dfrac{\partial^2\Theta(\vec{\psi})}{\partial\psi_i^2} = 0\ \forall i$. *The name, "multilinear polynomial", reflects the fact that the exponent of every* $\vec{\psi}$ *can only be either* 0 *or* 1.

Suppose that $\Theta(\vec{\psi})$ is a multilinear polynomial. For each $\vec{\psi} \in \Re^N$ and each index $i \in \{1, \ldots, N\}$, let $\vec{x_i}$ denote the $(N-1)$-dimensional vector obtained by omitting the i-th component of $\vec{\psi}$; that is:

$$\vec{x_i} = [\psi_1, \ldots, \psi_{i-1}, \psi_{i+1}, \ldots, \psi_N]^T. \tag{B.58}$$

Then there exist functions $E : \Re^{N-1} \longrightarrow \Re$ and $V : \Re^{N-1} \longrightarrow \Re$ such that:

$$\Theta(\vec{\psi}) = \psi_i E(\vec{x_i}) + V(\vec{x_i}). \tag{B.59}$$

Definition B.4.2 *A vector $\vec{\chi} \in \{-1, 1\}^N$ is said to be a local maximum of the objective function Φ if $\Phi(\vec{\chi}) \geq \Phi(\vec{y})$, $\forall\ \vec{y} \in \mathcal{N}(\vec{\chi})$, where $\mathcal{N}(\vec{\chi})$ denotes the set of all vectors in $\{-1, 1\}^N$ lying at a Hamming distance of one from $\vec{\chi}$. The vector $\vec{\chi}$ is said to be a strict local maximum of Φ if $\Phi(\vec{\chi}) > \Phi(\vec{y})$, $\forall\ \vec{y} \in \mathcal{N}(\vec{\chi})$.*

B.4.2 Parity Condition For Optimality

Proposition B.4.1 *Suppose that $\vec{\chi} \in \{-1, 1\}^N$ and $\Phi(\vec{\chi})$ is a multilinear polynomial. Then the following two statements are equivalent:*

1. *$\vec{\chi}$ is a local maximum of Φ.*
2. *$\vec{\chi}$ satisfies the parity condition:*

$$[\Phi(\vec{\chi})]_i \leq 0 \textit{ if } \chi_i = -1, \qquad [\Phi(\vec{\chi})]_i \geq 0 \textit{ if } \chi_i = 1. \tag{B.60}$$

Proof: "(2) $\Longrightarrow$ (1)" Suppose Statement 2 is true. Select an arbitrary index $i \in \{1, \ldots, N\}$, and define $\vec{y} \in \{-1, 1\}^N$ by:

$$y_i = -\chi_i,\ y_j = \chi_j,\ \forall\ j \neq i. \tag{B.61}$$

Then $\vec{y} \in \mathcal{N}(\vec{\chi})$, and $\vec{y_i} = \vec{x_i}$, where $\vec{x_i}$ is defined in (B.58), except here we have $\vec{\chi} \in \{-1, 1\}^N$ (as opposed to $\vec{\psi} \in \Re^N$). Hence, $E(\vec{y_i}) = E(\vec{x_i})$ and $V(\vec{y_i}) = V(\vec{x_i})$, and it follows from (B.59) that:

$$\Phi(\vec{y}) \;=\; y_i E(\vec{x_i}) + V(\vec{x_i}) = -\chi_i E(\vec{x_i}) + V(\vec{x_i}). \tag{B.62}$$

Therefore:

$$\Phi(\vec{\chi}) - \Phi(\vec{y}) \;=\; 2\chi_i E(\vec{x_i}). \tag{B.63}$$

Now, observe from (B.59) that:

$$\left[\vec{\nabla}_{\vec{\chi}} \Phi(\vec{\chi})\right]_i \;=\; \frac{\partial \Phi(\vec{\chi})}{\partial \chi_i} = \left[E(\vec{x_i})\right]_i. \tag{B.64}$$

Hence, if $\vec{\chi}$ satisfies the parity condition, (B.60), then the right hand side of (B.63) is always non-negative, and it follows that $\Phi(\vec{\chi}) \geq \Phi(\vec{y})$. Since the index, i, is arbitrary, the conclusion is that $\vec{\chi}$ is a local maximum.

"(1) $\Longrightarrow$ (2)" We show instead that if Statement 2 is false, then so is Statement 1. Suppose accordingly that the parity condition, (B.60), is violated for some index, i. Define $\vec{y}$ as in (B.61). Then the right hand side of (B.63) is negative for that i, which shows that $\Phi(\vec{\chi}) < \Phi(\vec{y})$, meaning that $\vec{\chi}$ is not a local maximum. ∎

Proposition B.4.2 *Suppose that $\vec{\chi} \in \{-1, 1\}^N$ and $\Phi(\vec{\chi})$ is a multilinear polynomial. Then the following two statements are equivalent:*

1. *$\vec{\chi}$ is a strict local maximum of Φ.*

2. *$\vec{\chi}$ satisfies the strict parity condition, that is, no component of $\Phi(\vec{\chi})$ is zero, and:*

$$\left[\Phi(\vec{\chi})\right]_i < 0 \textit{ if } \chi_i = -1, \qquad \left[\Phi(\vec{\chi})\right]_i > 0 \textit{ if } \chi_i = 1. \tag{B.65}$$

Proof: The proof is a routine modification of that of Proposition B.4.1. ∎

Appendix C

The Two-Morphogen Linear-Reaction Reaction-Diffusion Case Revisited

As promised in Section 3.2.3, we now solve the two-morphogen linear-reaction reaction-diffusion system exactly. For $M = 2$, we have:

$$\mathbf{R} = \begin{bmatrix} r_{11} & r_{12} \\ r_{21} & r_{22} \end{bmatrix}, \tag{C.1}$$

$$\mathbf{B}(\vec{k}) = \begin{bmatrix} B_1(\vec{k}) & 0 \\ 0 & B_2(\vec{k}) \end{bmatrix}, \tag{C.2}$$

and, consequently:

$$\mathbf{A}(\vec{k}) = \begin{bmatrix} r_{11} + B_1(\vec{k}) & r_{12} \\ r_{21} & r_{22} + B_2(\vec{k}) \end{bmatrix}. \tag{C.3}$$

Note that $B_1(\vec{k})$ and $B_2(\vec{k})$, the trigonometric polynomials in $\vec{k}$, have a finite number of terms, and thus correspond to FIR filters. When these filters represent diffusion, (C.3) reduces to the second equation in (3.6).

The eigenvalues of the linear-reaction reaction-diffusion system are:

$$\lambda_{1,2}(\mathbf{A}(\vec{k})) = \frac{\mathrm{trace}[\mathbf{A}(\vec{k})]}{2} \pm \left[\left(\frac{\mathrm{trace}[\mathbf{A}(\vec{k})]}{2}\right)^2 - \beta\right]^{\frac{1}{2}}, \tag{C.4}$$

$$\tau_e \stackrel{\mathrm{def}}{=} \mathrm{trace}[\mathbf{A}(\vec{k})] = \mathrm{trace}[\mathbf{R}] + \mathrm{trace}[\mathbf{B}(\vec{k})], \tag{C.5}$$

$$\beta \stackrel{\mathrm{def}}{=} \det[\mathbf{A}(\vec{k})] = \det[\mathbf{R}] + B_1(\vec{k})r_{22} + B_2(\vec{k})r_{11} + B_1(\vec{k})B_2(\vec{k}). \tag{C.6}$$

From (3.7), we conclude that for stability to homogeneous perturbations, (C.4) is subjected to the following constraints:

$$\mathrm{trace}[\mathbf{R}] < 0, \tag{C.7}$$

$$0 < \det[\mathbf{R}] < \left(\frac{\mathrm{trace}[\mathbf{R}]}{2}\right)^2, \tag{C.8}$$

$$\beta < \left(\frac{\mathrm{trace}[\mathbf{A}(\vec{k})]}{2}\right)^2. \tag{C.9}$$

Assume that $B_1(\vec{k})$ and $B_2(\vec{k})$ are real, *i.e.*, represent zero-phase FIR filters. Then all the relevant quantities (including the ones under the square root) are real. We are interested in determining the conditions, under which at least one of the eigenvalues becomes positive, while satisfying the constraints.

From (3.8), a positive eigenvalue can result in either or both of the following cases:

1. The trace of $\mathbf{A}(\vec{k})$ is positive, meaning that the trace of the matrix of the DFTs of the spatial filters is positive and has a greater magnitude than the trace of the reaction matrix:

$$\tau_e > 0. \tag{C.10}$$

2. The determinant of $\mathbf{A}(\vec{k})$ is negative:

$$\beta < 0. \tag{C.11}$$

C.1 Exact Linear Behavior

In this section, we continue exploring the properties of the two-morphogen linear-reaction reaction-diffusion system by finding its equivalent transfer function. We go back to (C.3) and write:

$$\mathbf{A}(\vec{k}) = \begin{bmatrix} a_{11}(\vec{k}) & a_{12}(\vec{k}) \\ a_{21}(\vec{k}) & a_{22}(\vec{k}) \end{bmatrix}. \tag{C.12}$$

We now expand the solution (3.14) for the two-morphogen case. Using the fact that:

$$\mathbf{A}(\vec{k}) = \mathbf{S}(\vec{k})\mathbf{\Lambda}(\vec{k})\mathbf{S}^{-1}(\vec{k}), \tag{C.13}$$

where $\mathbf{S}(\vec{k}) \in \Re^{2\times 2}$ is the matrix of the eigenvectors of $\mathbf{A}(\vec{k})$, and $\mathbf{\Lambda}(\vec{k}) \in \Re^{2\times 2}$ is the diagonal matrix of its eigenvalues:

$$\mathbf{\Lambda}(\vec{k}) = \begin{bmatrix} \lambda_1(\vec{k}) & 0 \\ 0 & \lambda_2(\vec{k}) \end{bmatrix},$$

we obtain [31]:

$$\exp\left[\mathbf{A}(\vec{k})t\right] = \mathbf{S}(\vec{k})\exp\left[\mathbf{\Lambda}(\vec{k})t\right]\mathbf{S}^{-1}(\vec{k}). \tag{C.14}$$

Define the determinant, the even trace, the odd trace, and the square root of the discriminant, respectively, as follows:

$$\beta \stackrel{\text{def}}{=} a_{11}a_{22} - a_{12}a_{21}, \tag{C.15}$$

$$\tau_e \stackrel{\text{def}}{=} a_{11} + a_{22}, \tag{C.16}$$

$$\tau_o \stackrel{\text{def}}{=} a_{11} - a_{22}, \tag{C.17}$$

$$\gamma \stackrel{\text{def}}{=} \sqrt{-4\,\beta + {\tau_e}^2}. \tag{C.18}$$

Then expanding (C.14) gives [1]:

$$\lambda_{1,2} = \frac{\tau_e \pm \gamma}{2}; \tag{C.19}$$

$$\mathbf{S}(\vec{k}) = \begin{bmatrix} \frac{-\gamma + \tau_o}{2a_{21}} & \frac{\gamma + \tau_o}{2a_{21}} \\ 1 & 1 \end{bmatrix}; \tag{C.20}$$

$$\mathbf{S}^{-1}(\vec{k}) = \begin{bmatrix} -\frac{a_{21}}{\gamma} & \frac{\gamma + \tau_o}{2\gamma} \\ \frac{a_{21}}{\gamma} & -\frac{(-\gamma + \tau_o)}{2\gamma} \end{bmatrix}; \tag{C.21}$$

$$\exp\left[\Lambda t\right] = \begin{bmatrix} \exp(\lambda_1 t) & 0 \\ 0 & \exp(\lambda_2 t) \end{bmatrix}; \tag{C.22}$$

$$\exp\left[\mathbf{A}(\vec{k})t\right] = \begin{bmatrix} \frac{(\gamma + \tau_o)e^{\lambda_1 t} + (\gamma - \tau_o)e^{\lambda_2 t}}{2\gamma} & \frac{(\gamma + \tau_o)(\gamma - \tau_o)\left(e^{\lambda_1 t} - e^{\lambda_2 t}\right)}{4a_{21}\gamma} \\ \frac{a_{21}\left(e^{\lambda_1 t} - e^{\lambda_2 t}\right)}{\gamma} & \frac{(\gamma - \tau_o)e^{\lambda_1 t} + (\gamma + \tau_o)e^{\lambda_2 t}}{2\gamma} \end{bmatrix}. \tag{C.23}$$

The input signal to the two-morphogen linear-reaction reaction-diffusion system is restricted in its functional form, because it has the meaning of the evocator (see Section 2.6.2). Hence, $\exp\left[\mathbf{A}(\vec{k})t\right]$ contains redundant information, which we now judiciously conceal. The evocator signal, $q(\vec{k})$, can be introduced to the system as one of three possible perturbations:

- Only the activator is perturbed: $\vec{\psi}(\vec{k}, t = 0) = \left[q(\vec{k})\ 0\right]^T$.
- Only the inhibitor is perturbed: $\vec{\psi}(\vec{k}, t = 0) = \left[0\ q(\vec{k})\right]^T$.
- Both morphogens are perturbed by the same signal: $\vec{\psi}(\vec{k}, t = 0) = \left[q(\vec{k})\ q(\vec{k})\right]^T$.

For all of these types of the initial condition, the transfer function is a 2-variable vector, but is constructed and used differently.

[1] In order to fit large equations onto the page, we use the notation $\exp(x)$ and e^x interchangeably.

C.1.1 Transfer Function With Both Morphogens Perturbed

We will begin by studying this particular case, because it requires a more detailed treatment than the other two cases and covers all the essential points of the analysis. The other cases, in which only one morphogen is perturbed by the evocator, are much simpler and will be analyzed subsequently.

In order to compute the transfer function (or the impulse response) of the linear-reaction reaction-diffusion system for this initial condition, we set the input to the unit sample so that $\vec{\Psi}(\vec{k}, t=0) = \vec{1}$. Hence, the transfer function is:

$$\begin{aligned}
\vec{H}(\vec{k}, t) &= \exp\left[\mathbf{A}(\vec{k})t\right] \begin{bmatrix} 1 \\ 1 \end{bmatrix} \\
&= \begin{bmatrix} \dfrac{(\gamma + \tau_o + 2a_{12}) \exp(\lambda_1 t) + (\gamma - \tau_o - 2a_{12}) \exp(\lambda_2 t)}{2\gamma} \\ \dfrac{(\gamma - \tau_o + 2a_{21}) \exp(\lambda_1 t) + (\gamma + \tau_o - 2a_{21}) \exp(\lambda_2 t)}{2\gamma} \end{bmatrix}. \qquad \text{(C.24)}
\end{aligned}$$

Now since for Turing's two-morphogen linear-reaction reaction-diffusion system $\tau_e < 0 \; \forall \; \vec{k}$, $\lambda_2 < 0$; therefore, the term, corresponding to λ_2, becomes negligible after a certain period of time. Hence:

$$\lim_{t \to \infty} \vec{H}(\vec{k}, t) = \begin{bmatrix} \dfrac{(\gamma + \tau_o + 2a_{12}) \exp(\lambda_1 t)}{2\gamma} \\ \dfrac{(\gamma - \tau_o + 2a_{21}) \exp(\lambda_1 t)}{2\gamma} \end{bmatrix}. \qquad \text{(C.25)}$$

The quantities $\lambda_{1,2}$, γ, β, τ_e, and τ_o in (C.25) depend on the spatial frequency $\vec{k}$. Therefore, (C.25) specifies a time-dependent filter. Figure C.1 shows the impulse response of the linear-reaction reaction-diffusion system.

It is interesting to note that the impulse response of the equivalent linear-reaction reaction-diffusion system is similar to that of the original non-linear Turing reaction-diffusion system, shown in Figure C.2. Moreover, if the range of morphogen concentrations is restricted to be small, then both systems respond similarly when the morphogen concentrations are initialized to random noise. We attribute this to the fact that Turing's reaction-diffusion system does not exhibit pronounced non-linear effects, unless the range of morphogen concentrations is large. See Figure C.3 and Figure C.4, respectively.

C.1.2 Transfer Function With Only Activator Perturbed

If the evocator is applied only to the activator morphogen, then for this initial condition we set the input to $\vec{\Psi}(\vec{k}, t=0) = [1\ 0]^T$. Hence, the transfer function is:

$$\begin{aligned}
\lim_{t\to\infty} \vec{H}(\vec{k}, t) &= \lim_{t\to\infty} \exp\left[\mathbf{A}(\vec{k})t\right] \begin{bmatrix} 1 \\ 0 \end{bmatrix} \\
&= \lim_{t\to\infty} \begin{bmatrix} \dfrac{(\gamma+\tau_o)\exp(\lambda_1 t) + (\gamma-\tau_o)\exp(\lambda_2 t)}{2\gamma} \\ \dfrac{a_{21}\left(\exp(\lambda_1 t) - \exp(\lambda_2 t)\right)}{\gamma} \end{bmatrix} \qquad \text{(C.26)} \\
&= \begin{bmatrix} \dfrac{(\gamma+\tau_o)\exp(\lambda_1 t)}{2\gamma} \\ \dfrac{a_{21}\exp(\lambda_1 t)}{\gamma} \end{bmatrix}. \qquad \text{(C.27)}
\end{aligned}$$

C.1.3 Transfer Function With Only Inhibitor Perturbed

If the evocator is applied only to the inhibitor morphogen, then for this initial condition we set the input to $\vec{\Psi}(\vec{k}, t=0) = [0\ 1]^T$. Hence, the transfer function is:

$$\begin{aligned}
\lim_{t\to\infty} \vec{H}(\vec{k}, t) &= \lim_{t\to\infty} \exp\left[\mathbf{A}(\vec{k})t\right] \begin{bmatrix} 0 \\ 1 \end{bmatrix} \\
&= \lim_{t\to\infty} \begin{bmatrix} \dfrac{(\gamma+\tau_o)(\gamma-\tau_o)\left(\exp(\lambda_1 t) - \exp(\lambda_2 t)\right)}{4a_{21}\gamma} \\ \dfrac{(\gamma-\tau_o)\exp(\lambda_1 t) + (\gamma+\tau_o)\exp(\lambda_2 t)}{2\gamma} \end{bmatrix} \qquad \text{(C.28)}
\end{aligned}$$

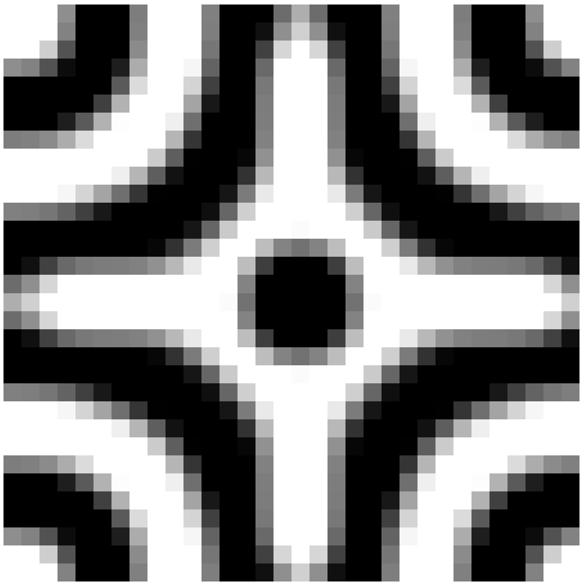


Figure C.1: Impulse response of the linear-reaction reaction-diffusion system that was derived from Turing's reaction-diffusion system ($N_x = N_y = 32$, $D_I = 6.25$).

$$= \begin{bmatrix} \dfrac{(\gamma + \tau_o)(\gamma - \tau_o)\exp(\lambda_1 t)}{4a_{21}\gamma} \\ \dfrac{(\gamma - \tau_o)\exp(\lambda_1 t)}{2\gamma} \end{bmatrix}. \tag{C.29}$$

C.2 Visual Appearance Of Morphogens

It has been mentioned in the literature that the plots of the activator and the inhibitor appear as the negatives of one another [4], [24]. This is demonstrated in Figure C.5. We now present an explanation of this phenomenon.

C.2.1 Activator And Inhibitor Out Of Phase

Consider a Turing linear-reaction reaction-diffusion system, *i.e.*, the system defined by (3.6) – (3.8). For $M = 2$, the relevant equations are (C.1) – (C.29). According

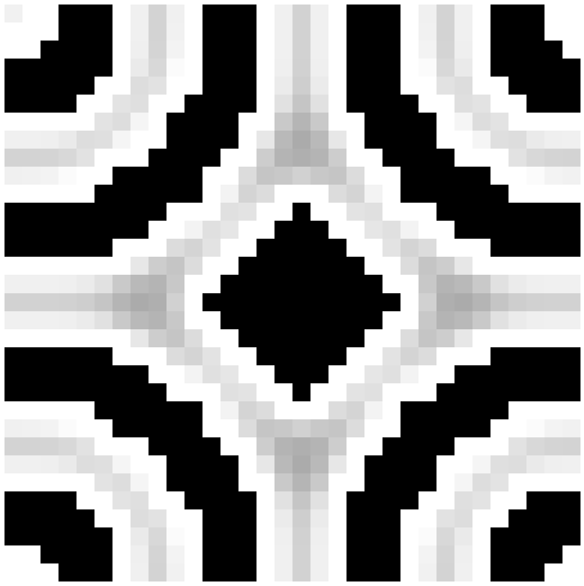
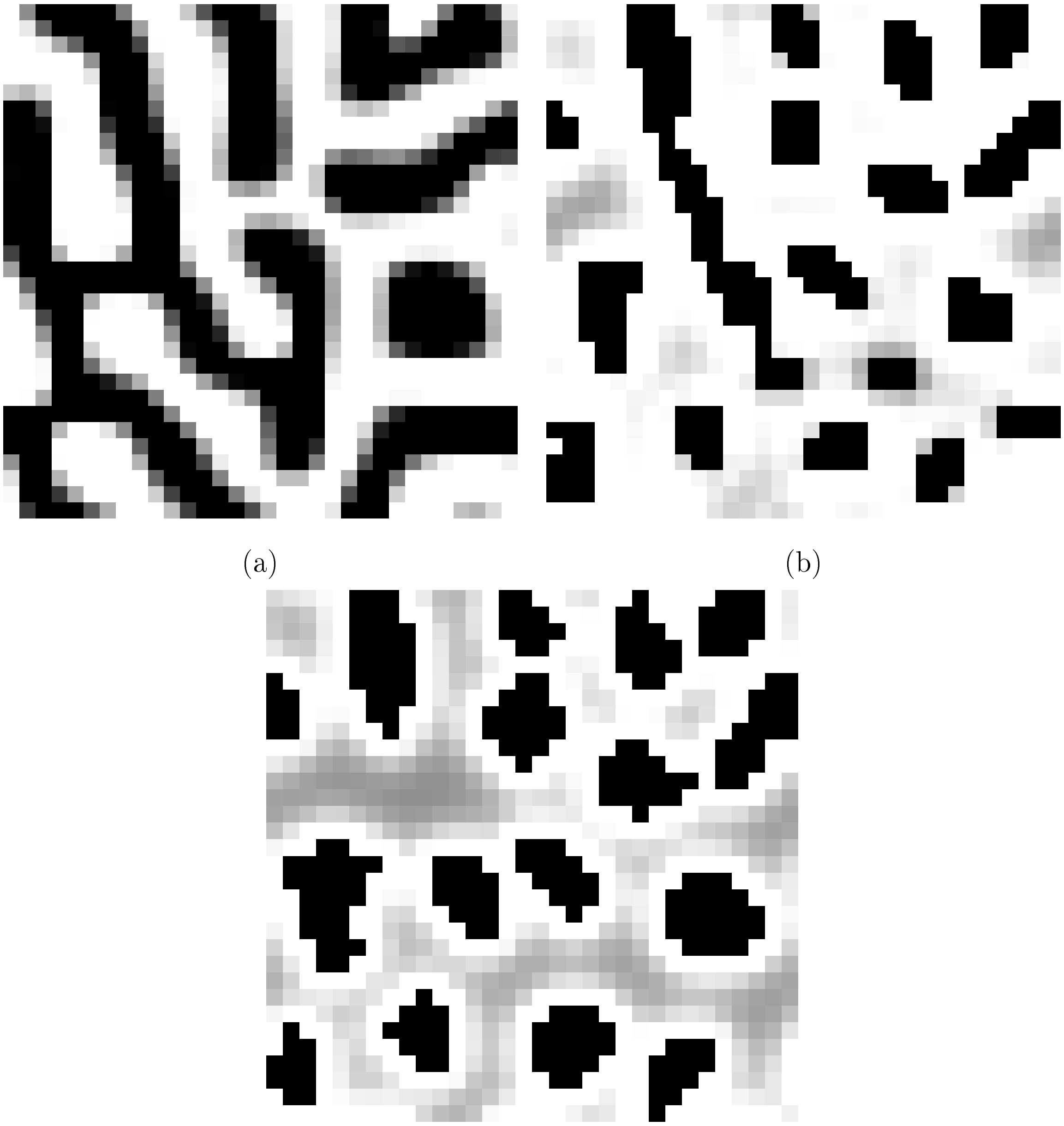


Figure C.2: Impulse response of Turing's reaction-diffusion system ($N_x = N_y = 32$, $D_I = 6.25$).

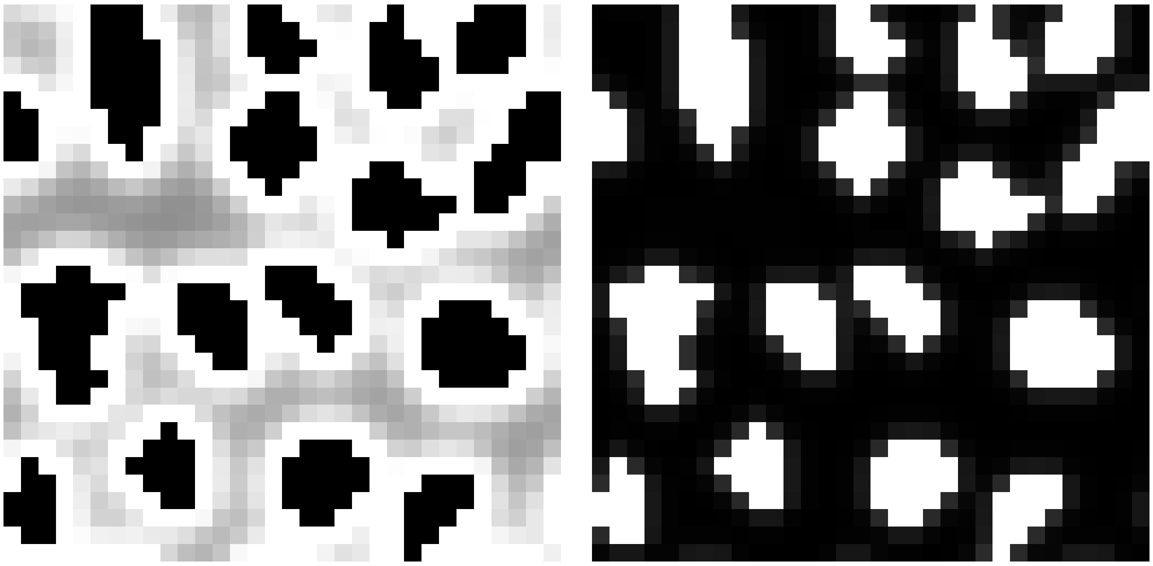
Figure C.3: The response of the linear-reaction reaction-diffusion system to random noise ($N_x = N_y = 32$, $D_I = 6.25$).



(a) (b)

(c)

Figure C.4: The response of Turing's reaction-diffusion system to random noise ($N_x = N_y = 32$, $D_I = 6.25$): (a) the range of morphogen concentrations is 2; (b) the range of morphogen concentrations is 20; (c) the range of morphogen concentrations is 200.



(a) (b)

Figure C.5: The visual appearance of the plots of the activator and the inhibitor concentrations of Turing's reaction-diffusion model. The two morphogens appear to be the negatives of one another ($N_x = N_y = 32$, $D_I = 6.25$): (a) the activator morphogen; (b) the inhibitor morphogen.

to these equations, in Turing's two-morphogen linear-reaction reaction-diffusion system at least one of the diagonal elements of $\mathbf{R}$ must be negative; otherwise, the trace of $\mathbf{R}$ will not be negative. Assume also that $B_1(\vec{k})$ and $B_2(\vec{k})$ represent the standard diffusion, in which case both $B_1(\vec{k})$ and $B_2(\vec{k})$ are non-positive. Then at least one of the diagonal elements of $\mathbf{R}$ must be positive; otherwise, the determinant of $\mathbf{A}(\vec{k})$ will never be negative, thereby precluding the diffusional instability. Therefore, a necessary condition for a Turing two-morphogen linear-reaction reaction-diffusion system is that the diagonal elements of $\mathbf{R}$ have the opposite signs.

Both Morphogens Perturbed

Assume that $r_{11} > 0$ and $r_{22} < 0$. Then $r_{11} - r_{22} > 0$. Noting that in the two-morphogen linear-reaction reaction-diffusion system the diffusion rates of the inhibitor are typically much larger than those of the activator ($D_{2,x} \gg D_{1,x}$, $D_{2,y} \gg D_{1,y}$), we use the definition of τ_o to realize that $\tau_o \gg 0$. Then, using the definition of τ_e, we realize that $\tau_e \ll 0$. Assume for the moment that $r_{12} \gg 0$ and $r_{21} \ll 0$. Then, because $r_{11} - r_{22} > 0$, τ_o becomes sufficiently large to offset the effect of τ_e and $\beta < 0$ in (C.25). Together with our assumption on the mutual reaction rates, r_{12} and r_{21}, this dictates that the spatial frequencies with a positive eigenvalue will grow in the positive direction for the activator and in the negative direction for the inhibitor. In other words, the signs of the amounts of these spatial frequencies are opposite for the activator and the inhibitor. In the spatial domain, this is equivalent to the fact that the corresponding sinusoidals for the activator and the inhibitor are 180° out of phase and have different amplitudes.

Note that the dependence on the spatial frequency of the quantity under the square root in (C.25) does not allow the activator and the inhibitor to be exactly 180° out of phase (*i.e.*, to have the amplitudes that are equal in magnitude and opposite in sign). To see this, consider the following equation:

$$a_{12} + a_{21} \quad = \quad -\sqrt{-4\,\beta + \tau_e^2}, \tag{C.30}$$

which is derived from (C.25) by setting $H_1(\vec{k}, t) = -H_2(\vec{k}, t)$. Clearly, it cannot be satisfied with constant reaction rates.

See Section 2.4.4 of Chapter 2) for the actual marginal reaction rates of Turing's two-Morphogen reaction-diffusion system.

Only Activator Perturbed

The preceding discussion establishes that $\gamma > 0$ and $\tau_o > 0$. Also note that $a_{21} = r_{21} < 0$. Then according to (C.27), the two components of $\vec{H}(\vec{k}, t)$ have opposite signs. This can be seen more clearly by realizing that the first component of $\vec{H}(\vec{k}, t)$ differs from the second by the factor of $\dfrac{\gamma + \tau_o}{2a_{21}}$, which is a negative quantity. Hence, the two morphogen concentrations evolve in the opposite directions.

Only Inhibitor Perturbed

According to (C.29), the first component of $\vec{H}(\vec{k}, t)$ is $\dfrac{\gamma + \tau_o}{2a_{21}}$ times the second, as in the previous section. Therefore, the two morphogen concentrations evolve in the opposite directions under this set of initial conditions also.

C.2.2 Activator And Inhibitor In Phase

Alternatively, assume that $r_{12} \ll 0$ and $r_{21} \gg 0$. Then it is possible to have a Turing linear-reaction reaction-diffusion system, in which both the activator and the inhibitor are in phase, but with different amplitudes. This can be seen by examining (C.25), (C.27), and (C.29) again. One can make $H_1(\vec{k}, t)$ positive if the positive quantities, β and τ_o, outweigh the negative quantity, r_{12}. Similarly, one can make $H_2(\vec{k}, t)$ also positive if the positive quantities, β and r_{21}, outweigh the negative quantity, $-\tau_o$. Again, $H_1(\vec{k}, t)$ and $H_2(\vec{k}, t)$ cannot have equal magnitude.

Appendix D

General Noise-Shaping System

This appendix reviews the theory of noise shaping. For many years, researchers in the oversampled Σ-Δ modulators for high-fidelity audio [79] and researchers in the error-diffusion algorithms for high-quality printing [80] had been working on the same general noise-shaping concepts without realizing it [73]. Even though the (1-D) audio and the (2-D) image domains cause the actual implementations to differ substantially in practice, both types of the system essentially aim to binarize the input signal in a way that places the useful information and the quantization error into the separate portions of the frequency spectrum. In particular, the goal is to make the quantization noise perceptually unnoticeable. We now develop this theory for the use in Chapter 5.

Consider the discrete system, shown in Figure D.1(a). Its operation can be analyzed quantitatively by modeling the quantization process by an additive noise source, $Q(\vec{z})$, as illustrated in Figure D.1(b). For images, $\vec{z} \in \mathcal{C}^2$. The noise source, $Q(\vec{z})$, is assumed to be white and statistically uncorrelated, which has been shown to be a reasonable assumption for non-DC signals [84], [83], [81]. With this linearized model of the system, it can be shown that:

$$Y(\vec{z}) = H_S(\vec{z})S(\vec{z}) + H_Q(\vec{z})Q(\vec{z}). \tag{D.1}$$

This equation states that the spectrum of the output (binary) image consists of two components: the input signal filtered by the input transfer function and the noise signal filtered by the noise transfer function. This illustrates the noise-shaping attribute of the system in Figure D.1. The generality of this system can be illustrated by deriving from it the error diffusion and the oversampled Σ-Δ modulator architectures.

D.1 Deriving Error Diffusion

For error diffusion, we want to pass the input signal unaltered and give the quantization noise the desired "blue" character [69], [93]. This means that in (D.1) $H_S(\vec{z}) = 1$, and $H_Q(\vec{z})$ is a high-pass filter, such as the one shown in Figure D.2. Note that the magnitude of the filter must never exceed unity for the reasons of stability. Observing that

$$Q(\vec{z}) = Y(\vec{z}) - A(\vec{z}), \tag{D.2}$$

we substitute (D.2) into (D.1) and obtain:

$$\begin{aligned} Y(\vec{z}) &= S(\vec{z}) + H_Q(\vec{z})(Y(\vec{z}) - A(\vec{z})), \\ (Y(\vec{z}) - A(\vec{z})) + A(\vec{z}) &= S(\vec{z}) + H_Q(\vec{z})(Y(\vec{z}) - A(\vec{z})), \\ A(\vec{z}) &= S(\vec{z}) + (H_Q(\vec{z}) - 1)(Y(\vec{z}) - A(\vec{z})), \quad \text{or} \qquad \text{(D.3)} \\ A(\vec{z}) &= S(\vec{z}) - (1 - H_Q(\vec{z}))(Y(\vec{z}) - A(\vec{z})). \qquad \text{(D.4)} \end{aligned}$$

Clearly, in either case of (D.3) or (D.4) the magnitude of the error filter,

$$\begin{aligned} F_-(\vec{z}) &= H_Q(\vec{z}) - 1 \quad \text{or} \qquad \text{(D.5)} \\ F_+(\vec{z}) &= 1 - H_Q(\vec{z}), \qquad \text{(D.6)} \end{aligned}$$

exhibits the low-pass behavior. Since the magnitude of the noise shaping filter, $H_Q(\vec{z})$,

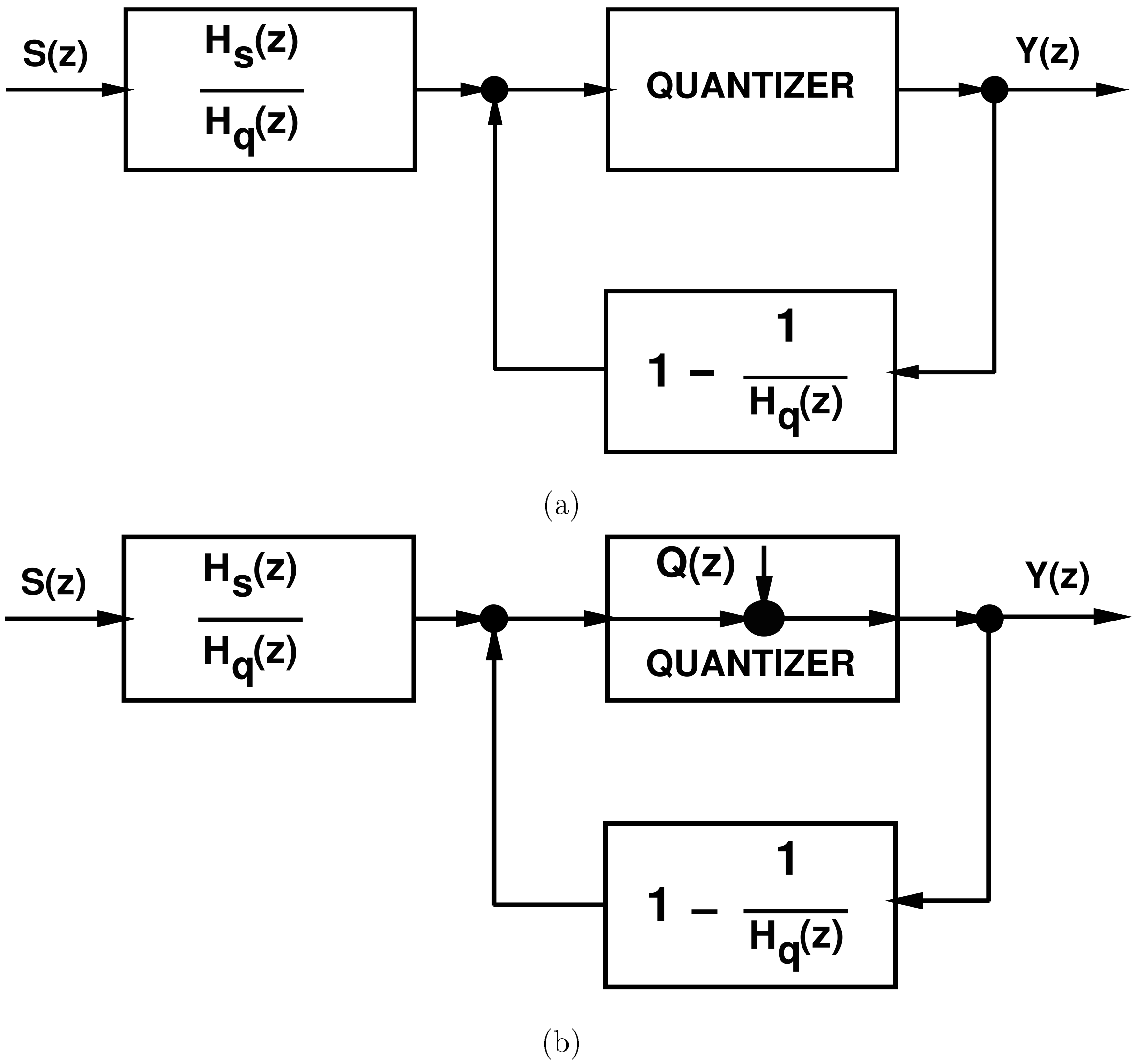


Figure D.1: Block diagram of the general noise-shaping system: (a) the actual (non-linear) system; (b) the linearized model of the system in (a).

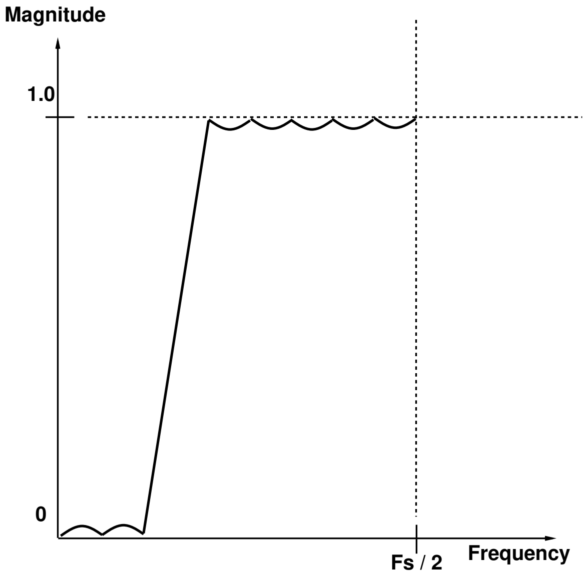


Figure D.2: A high-pass noise-shaping filter.

obeys

$$0 \leq H_Q(\vec{z}) \leq 1, \tag{D.7}$$

we obtain the following restrictions on the error diffusion filter:

$$-1 \leq F_-(\vec{z}) \leq 0, \quad \text{or} \tag{D.8}$$

$$0 \leq F_+(\vec{z}) \leq 1, \tag{D.9}$$

depending on the choice of representation format. These restrictions stem from the requirement that the effective noise-shaping filter, $H_Q(\vec{z})$, must remain stable.

D.2 Deriving Oversampled Σ-Δ Modulator

For oversampled Σ-Δ modulators, in addition to (D.1) it is required that the signal and the noise filters complement one another in the frequency domain [84]:

$$H_S(\vec{z}) + H_Q(\vec{z}) = 1. \tag{D.10}$$

Again, we want the quantization noise to be blue. This means $H_Q(\vec{z})$ is a high-pass filter, and $H_S(\vec{z})$ is its low-pass complement. The system acts as a low-pass filter on the signal and as a high-pass filter on the noise. Substituting (D.10) into (D.1) and denoting the low-pass signal filter by $F(\vec{z})$, gives:

$$\begin{aligned}
Y(\vec{z}) &= F(\vec{z})S(\vec{z}) + (1 - F(\vec{z}))Q(\vec{z}) \\
&= F(\vec{z})S(\vec{z}) + (1 - F(\vec{z}))(Y(\vec{z}) - A(\vec{z})) \\
&= (Y(\vec{z}) - A(\vec{z})) + A(\vec{z}), \\
A(\vec{z}) &= F(\vec{z})S(\vec{z}) - F(\vec{z})(Y(\vec{z}) - A(\vec{z})), \\
A(\vec{z}) &= \left(\frac{F(\vec{z})}{1 - F(\vec{z})}\right)(S(\vec{z}) - Y(\vec{z})) \,. \qquad \text{(D.11)}
\end{aligned}$$

Typically, block diagrams of oversampled Σ-Δ modulators are constructed using (D.11).

Appendix E

Least-Squares Halftoning

This appendix reviews two popular methods for the digital halftoning of images. Both of these methods are employed in Chapter 5.

Suppose $\vec{n} \in \mathcal{Z}^2$; $s(\vec{n}) \in [-1, 1]$ is the continuous-tone (or finely-quantized) original input image signal; $y(\vec{n}) \in \{-1, 1\}$ is the output halftone image; and $h(\vec{n})$ is a 2-D low-pass filter, such as, for example, the one suggested by Mannos and Sakrison [71]. Then the problem of halftoning can be stated as a non-linear program.

E.1 Filtered-Squared-Error Method

The approach of minimizing the unfiltered squared error between the original and the halftoned images leads to the fixed-thresholding technique, which produces halftoned images that are visually unsatisfactory [75], [78]. An alternative is to use a frequency-weighted (or filtered) squared error, which has better visual properties than the unfiltered squared error [73], [94]. Minimize:

$$E(y(1,1),\ \ldots,\ y(N_x,N_y)) = \frac{1}{2}\sum_{\vec{n}} \left[h(\vec{n}) * (s(\vec{n}) - y(\vec{n}))\right]^2 \tag{E.1}$$

subject to constraints:

$$y^2(\vec{n}) - 1 \geq 0. \tag{E.2}$$

In a matrix-vector form, this can be restated as:

$$\begin{aligned}
\min_{\vec{y}} D(\vec{s},\vec{y}) &= \min_{\vec{y}} \frac{1}{2}\left[\mathbf{H}(\vec{s}-\vec{y})\right]^T\left[\mathbf{H}(\vec{s}-\vec{y})\right] && \text{(E.3)}\\
&= \min_{\vec{y}} \frac{1}{2}(\vec{s}-\vec{y})^T(\mathbf{H}^T\mathbf{H})(\vec{s}-\vec{y}) && \text{(E.4)}\\
&= \min_{\vec{y}} \frac{1}{2}(\vec{s}-\vec{y})^T\mathbf{B}(\vec{s}-\vec{y}) && \text{(E.5)}\\
&= \min_{\vec{y}} \frac{1}{2}\vec{y}^T\mathbf{B}\vec{y} - (\mathbf{B}\vec{s})^T\vec{y} && \text{(E.6)}
\end{aligned}$$

subject to constraints:

$$y_i^2 - 1 \geq 0, \tag{E.7}$$

where the vectors are the lexicographical concatenations of the corresponding sequences, $\mathbf{H}$ is the matrix representation of $h(\vec{n})$, and $\mathbf{B} = \mathbf{H}^T\mathbf{H}$ [1]. The particular form of constraints, (E.7), forces each pixel to assume binary values.

[1]Note that $b(\vec{n}) = h(-\vec{n}) * h(\vec{n})$, the auto-correlation of $h(\vec{n})$.

E.2 Least-Squares Intensity-Approximation Method

This approach aims to calculate a bi-level image, whose average intensity mimics the intensity of the original continuous-tone image [95]:

$$\begin{aligned}
\min_{\vec{y}} D(\vec{s}, \vec{y}) &= \min_{\vec{y}} \frac{1}{2} [\vec{s} - \mathbf{H}\vec{y}]^T [\vec{s} - \mathbf{H}\vec{y}] && \text{(E.8)} \\
&= \min_{\vec{y}} \left\{ -(\mathbf{H}^T \vec{s})^T \vec{y} + \frac{1}{2} \vec{y}^T \mathbf{B} \vec{y} \right\} && \text{(E.9)}
\end{aligned}$$

subject to constraints:

$$y_i^2 - 1 \geq 0, \tag{E.10}$$

where the averaging is performed by $h(\vec{n})$ (represented by $\mathbf{H}$), and $\mathbf{B} = \mathbf{H}^T\mathbf{H}$.

The "energy" function in (E.9) can be interpreted in the following way. The first term indicates that the original continuous-tone image acts as an external field. The total energy is reduced if the binary pixels align with the external field (*i.e.*, acquire the same polarity as the corresponding pixels in the original image). In other words, the field term gives the binary image the tendency to resemble the original continuous-tone image. On the other hand, the second (neighborhood) term reduces the total energy if the polarity of every pixel is opposite from that of its neighbors. This tendency facilitates high-frequency rendition and noise shaping.